M&S JOURNAL

A DoD MODELING & SIMULATION COORDINATION OFFICE PUBLICATION | SPRING 2012

The Tools Issue

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **JUN 2012**	2. REPORT TYPE	3. DATES COVERED **00-00-2012 to 00-00-2012**
4. TITLE AND SUBTITLE **M&S Journal. Spring 2012**		5a. CONTRACT NUMBER
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **DoD Modeling & Simulation Office,1901 N Beauregard St Ste 500,Alexandria,VA,22311-1705**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**72**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

FROM THE EXECUTIVE EDITOR

THE UNDER SECRETARY OF DEFENSE for Acquisition, Technology, and Logistics (AT&L) and the Modeling and Simulation Coordination Office (M&SCO) are responsible for improving the effectiveness and cost-effectiveness of M&S across the Department of Defense (DoD). To accomplish these goals, we use an enterprise approach that is focused on re-use and interoperability, while reducing unnecessary duplication. The Tools Issue of the M&S Journal explores several aspects of this approach.

■ The article on LVC architecture presents the need for including an enterprise approach to identifying unnecessarily duplicative tools and ways to streamline future architectures. ■ The CREATE article discusses leveraging academic and commercial developments in high performance computing to increase the effectiveness of M&S tools for the DoD M&S enterprise. ■ The paper on effective training games offers insight into using open source gaming engines as the backbone of important training tools and indicates how open sources provide a path for increased interoperability and re-use. ■ The data logging paper summarizes a vital adjunct tool based on simulation architecture standards, including those developed by AT&L. ■ The article on ship design illustrates the need to integrate M&S tools for the full system acquisition lifecycle and confirms that Simulation Based Design (SBD) remains a viable concept to improve interoperability and cost-effectiveness of DoD systems.

Using an enterprise approach to funding tools in DoD allows us to counteract the M&S adage that "a tool and its money are soon parted." Continued funding for enterprise tools is necessary to ensure they (1) are applicable and usable, (2) remain visible and available, and (3) are sustained to maximize re-use. Please visit the M&SCO website at http://www.msco.mil/ for more information on our approach to enterprise tools, data, and services, and other DoD M&S enterprise activities, and to download digital versions of the M&S Journal.

J. DAVID LASHLEE, PH.D., CMSP

Deputy Director
Modeling and Simulation Coordination Office (M&SCO)

TABLE OF CONTENTS

GUEST EDITORIAL

Frank Mullen
Associate Director for Tools, M&SCO

Mankind have been making tools since Ogg first chipped an edge on flint. It is characteristic of our species that we not only use tools—some other animals use them too, I'm told—but we conceive of and form them to our purposes.

In a simpler day, tools were tangible things. Hold it in your hand and pound a nail? Hammer: tool. Climb into it and move a pile of dirt? Bulldozer: tool. Easy to understand. Nowadays, with so much being done on computers, tools are harder to grasp. Modeling and simulation tools are typical in this regard. I have been asked on a number of occasions, as the DoD Modeling and Simulation Coordination Office's (M&SCO) Associate Director for Tools, to provide a definition of M&S tools. Perhaps because the work they do exists in virtual space, they are intrinsically hard to recognize. You generally cannot see one, or hold it, touch it, or climb into it. Even when the user interface is tangible—cockpit simulators and machine-gun trainers come to mind—there is a computer in the background running sophisticated software and processing great volumes of data, all unseen to the user, and completely intangible.

I will not provide here a definition of "tools" for M&S. The general understanding of the term works well enough, and were I to attempt it, someone would be sure to bring to my attention some manifestly useful tool that nevertheless failed the definition. If someone conceived of it and built it to purpose, if it is useful for designing, building, operating, or otherwise pushing along a model or simulation, then surely it is an M&S tool.

The variety of tools is enormous. A glance through this issue of the M&S Journal illustrates the broad scope of M&S tools. Examples here address the range from particular problems—see Talib S. Hussain, et al., "Designing and Developing Effective Training Games for the U.S. Navy"—to tools of broad applicability, as in Björn Möller, et. al., "Scalable and Embeddable Data Logging for Live, Virtual and Constructive Simulation: HLA, Link 16, DIS and more."

But the very broad scope of tool applicability also suggests a definition for a particular type of M&S tool, the "enterprise tool." Within the DoD it's not just any broadly-applicable M&S tool: the USD (AT&L) designates such tools after consultation with the M&S Steering Committee. The M&S enterprise tools can be accessed from the M&SCO website: http://www.msco.mil/tools.html.

You will find here that there are so far only a few M&S enterprise tools. They include:

- The M&S Catalog
- The Standards Vetting Tool (SVT)
- The VV&A Documentation Tool (VDT)
- Capabilities Requirements Tool (CRT).

These enterprise tools were developed individually, but M&SCO has begun the process of gathering requirements from users to integrate them. Subject matter experts from the Services and the Communities enabled by M&S have been invited to participate in this effort, which began in February of this year.

The purpose of the integration effort is to facilitate discovery and reuse of M&S assets. This advances the goals of the "Strategic Vision for DoD Modeling and Simulation" issued by the OUSD (AT&L) and the M&S Steering Committee, as well as the "DoD Net-Centric Data Strategy.[i]" A previous contributor[ii] to this column likened the current state of discovery and reuse to a "farmer who goes out and buys a shiny new harvester, but forgets to grow the hay. The mowing gets done faster—but there's still not much hay." By this statement he meant that the M&S enterprise has a metadata catalog capable of making M&S assets far more visible.

There are multiple approaches to populating the catalog with more useful metadata. Policy, for example, can be developed to emphasize the value of discovery and reuse to the DoD. Another approach is to look for ways to provide value to M&S users and developers and to harvest metadata as a byproduct. The enterprise tools integration effort is starting up. The idea is to provide, through the enterprise tools, services that users and developers will take advantage of because of their perceived value, e.g., in reducing labor costs or making documentation less onerous. At the same time, an integrated set of enterprise tools will automatically harvest metadata, make it visible in the catalog, and keep it updated. In the future, this set of enterprise tools could make M&S assets visible and reusable throughout their lifecycle, from initial definition of their requirements (e.g., through CRT) through delivery and acceptance (e.g., through VDT), to operation and sustainment.

It appears that I have, despite myself, provided a definition of at least a small number of M&S tools. They are, as are all tools, conceived of and made for a purpose: to aid discovery and reuse. They are designed to provide a service to users while at the same time delivering value to the Department.

REFERENCES

[i] Strategic Vision: http://www.msco.mil/strategicVision.html Net-Centric Data Strategy: http://dodcio.defense.gov/docs/Net-Centric-Data-Strategy-2003-05-092.pdf

[ii] John W. Diem, Guest Editorial, M&S Journal, Fall 2011

ABOUT THE AUTHOR

FRANK MULLEN is the Associate Director for tools at M&SCO. He was for many years on the technical staff of the Charles Stark Draper Laboratory in Cambridge, MA. He is a graduate of the US Coast Guard Academy, the US Naval War College, and the California Institute of Technology.

LVC ARCHITECTURE ROADMAP IMPLEMENTATION – *results of the first two years*

James E. Coolahan, Ph.D.
Johns Hopkins University Applied Physics Laboratory
11100 Johns Hopkins Road
Laurel, MD 20723-6099
240-228-5155
James.Coolahan@jhuapl.edu

Gary W. Allen, Ph.D.
Joint Training Integration and Evaluation Center
12000 Research Parkway, Suite 300
Orlando, FL 32826
407-208-5607
gary.allen@us.army.mil

ABSTRACT

The implementation of recommendations from the Live-Virtual-Constructive Architecture Roadmap (LVCAR), which was performed under the auspices of the U.S. Office of the Secretary of Defense (OSD), was begun in mid-2009. Under the leadership of the Joint Training Integration and Evaluation Center (JTIEC), the Johns Hopkins University Applied Physics Laboratory (JHU/APL) has undertaken multiple implementation efforts in the areas of common capabilities for LVC simulations, gateways for multi-architecture LVC simulations, and convergence of LVC simulation architectures. A number of papers presented at Simulation Interoperability Workshops since the spring of 2010 have described individual activities that are part of this overall effort.

This paper provides a comprehensive summary of the results of the first two years of LVCAR Implementation (LVCAR¬I) efforts. It describes accomplishments in the development of new prototype standards for the multi-architecture simulation systems engineering process and for multi-architecture simulation federation agreements, as well as a tool to aid in the implementation of such federation agreements. It also discusses candidate business models to enhance the potential for reuse of LVC simulation tools, and pilot efforts to explore the feasibility of such business models. Mechanisms for describing LVC simulation assets using standardized metadata are described, in conjunction with the development of a prototype implementation of a portal for discovering and locating such assets for subsequent download for reuse. Additionally, storage formats for LVC simulation-related data are categorized, along with opportunities for improved commonality. Advances in the description and characterization of simulation gateways are also provided that will permit more informed selection of gateways by users for particular applications.

With respect to current and future trends in Modeling and Simulation (M&S) technology, the paper describes efforts related to Service-Oriented Architectures (SOAs) and identifying future technologies having potential use for the DoD Modeling and Simulation Community. Finally, the paper provides an overview of the way ahead for the next two years of the LVCAR implementation effort.

KEYWORDS

Live-Virtual-Constructive Simulation, Simulation Standards, Multi-Architecture Simulation Gateways, Common Simulation Data Formats, Model & Simulation Asset Reuse

AUTHOR BIOGRAPHIES

DR. JAMES E. COOLAHAN is a Program Manager for Modeling and Simulation, and a National Security Studies Fellow, in the National Security Analysis Department at the Johns Hopkins University Applied Physics Laboratory (JHU/APL), where he also served as the Assistant to the Director for Modeling and Simulation from 1996 to 2001. In his 39 years at JHU/APL, his technical activities have included modeling and simulation, test and evaluation, and the development of oceanographic data acquisition systems. Dr. Coolahan served as a member of the National Research Council (NRC) Committee on Modeling and Simulation Enhancements for 21st Century Manufacturing and Acquisition from 2000 to 2002. He currently chairs the Modeling and Simulation Committee of the Systems Engineering Division of the National Defense Industrial Association (NDIA), serves as secretary of the Conference Committee of SISO, and is a member of the I/ITSEC Tutorial Board. Dr. Coolahan holds B.S. and M.S. degrees in aerospace engineering from the University of Notre Dame and the Catholic University of America, respectively, and

M.S. and Ph.D. degrees in computer science from the Johns Hopkins University and the University of Maryland, respectively.

DR. GARY W. ALLEN has worked in various aspects of modeling and simulation for the past 30 years. He was a member of the team that founded the Training Simulation Center for I Corps at Ft. Lewis, Washington (1980), Director of the Simulation Training Branch at the US Army Intelligence Center and School, Ft. Huachuca, AZ (1989-1992), and Project Director for the TACSIM Intelligence Simulation and part of the design group that initiated the Aggregate Level Simulation Protocol (ALSP). From 1996 – 2008, he was the US Army Liaison Officer to the German Military Research and Development Agency in Koblenz, Germany. Currently he is Project Manager for the DoD High Level Task, "Live, Virtual, Constructive Architecture Roadmap Implementation" project. His academic background includes a 1978 MS in Telecommunications Systems Management, School of Engineering, University of Colorado (Boulder), and a 1989 PhD in Instructional Technology, University of Kansas (Lawrence). Dr. Allen is a member of the Phi Kappa Phi National Honor Society, a 1999 graduate of the Army War College, and an Acquisition Corps Level III Certified PM.

1. BACKGROUND

The Live-Virtual-Constructive Architecture Roadmap Implementation (LVCAR-I) and Net-centric Environment project is the follow-on effort that stems from findings of the LVCAR Final Report [1]. The purpose of the LVCAR study was to: "Develop a future vision and supporting strategy for achieving significant interoperability improvements in LVC simulation environments." The focus of the study revolved around four dimensions of simulation interoperability: Technical Architecture, Business Models, Standards Evolution, and Management Processes. The precepts that guide the study's implementation are: do no harm, interoperability is not free, start with small steps, and provide central management. Among the significant results of the LVCAR study is a set of 19 recommendations. These recommendations act as the requirements document found in formal programs and is used to guide the LVCAR-I tasks.

The principal aims of LVCAR-I are to explore organizational and structural (e.g., use of standards) options to better:

1. manage LVC architecture interoperability;
2. create reference models to focus data and service reuse efforts;
3. reduce LVC architecture divergence and tool proliferation; and
4. explore emerging technology issues related to future LVC architecture performance and requirements.

The planning, development, and execution of LVC events are universally recognized to require an investment of resources. Also, the M&S community has limited agility with regards to supporting unforeseen events. Given this situation, the objective of LVCAR-I is to reduce overhead and provide for an interoperable M&S environment, thus improving the ability to construct and conduct timely LVC events. Described another way, the goal for LVCAR-I is to get M&S support inside the military operations decision cycle.

The project leads have taken a holistic approach to organization and definition of an acquisition strategy. Fundamentally, LVCAR-I is designed to work in an environment where there are many different factors and incentives that influence decisions, including willingness to change and the adoption of technical solutions. Understanding these factors and their effects are as important to the success of the project as the technology advances themselves. As a result, the LVCAR-I team distilled the 19 recommendations found in the LVCAR study to the grouping of core, affiliated, and supporting efforts as described in Table 1.

2. OVERVIEW OF THE LVCAR IMPLEMENTATION EFFORT

In addition to the 19 LVCAR recommendations being grouped as shown in Table 1, the technical efforts being performed as part of LVCAR-I were subdivided into four major technical areas:

1. LVC Common Capabilities;
2. LVC Gateways and Bridges;
3. LVC Architecture Convergence; and
4. LVC Future-Oriented Efforts.

Within the LVC Common Capabilities technical area, efforts were further subdivided into:

a. Systems Engineering Process

b. Federation Agreements

c. Reusable Tools and Common Data Storage Formats

d. Asset Reuse Mechanisms.

	Core Task	Affiliated Task	Supporting Task
Standards Development	Systems Engineering Process		
	Federation Agreement Templates		
	Reusable Development Tools		
	Asset Reuse Mechanisms		
Software Development	Common Gateways & Bridges	Joint Composable Object Model	
	Architecture Convergence		
Studies	Management – Product Transition Strategy	Management Organizations and Processes	SOA Concepts
			LVC Futures
Outreach	Core Task Workshops	Management Workshops	M&S Forums Presentations
			Working Group Presentations
			Web-based Information

Table 1. Overview of LVCAR-I Efforts

Within the LVC Future-Oriented Efforts technical area, efforts were further subdivided into:

a. Service-Oriented Architecture (SOA) Application to LVC Simulations; and

b. LVC Futures.

From a functional perspective, however, these technical areas can be reformulated into several major objectives:

1. Prototyping LVC Simulation Standards;
2. Advancing the Reuse of LVC Simulation Assets;
3. Increasing the Commonality of Data Storage Formats;
4. Improving the Use of Gateways and Bridges for LVC Simulations;
5. Investigating LVC Architecture Convergence; and
6. Investigating the Application of Additional Technologies to LVC Simulations.

The following sections discuss each of the above objectives, and the progress made to date in achieving them in the first two years of LVCAR implementation. In most cases, these individual efforts have been documented in technical papers that have been previously presented at the semi-annual Simulation Interoperability Workshops (SIWs) and the annual Interservice/Industry Training, Simulation & Education Conference (I/ITSEC), as well as other venues. For example, a full-day workshop on the initial progress of the effort was conducted at the 2010 Spring SIW [2] to get feedback from the broader M&S community. The purpose of this paper is not to repeat prior papers in detail, but rather to present a consolidated summary of the first two years of the project that can be used as a reference point for further exploration of the more detailed efforts. Citations of publicly available technical papers on the more detailed efforts are made, where appropriate. Project-specific technical reports on the various efforts are available through appropriate program channels.

3. PROTOTYPING LVC SIMULATION STANDARDS

Although simulation standards, to gain widespread acceptance within the community, need to be developed using a consensus-based process, it is sometimes necessary to "seed" the development of such standards by undertaking a funded effort to create a prototype upon which subsequent volunteer efforts can be based. In the area of LVC simulations, it was felt, based on the LVCAR study, that there were two areas in which such prototype efforts were needed:

- A Multi-Architecture Systems Engineering Process for LVC Simulations; and
- An LVC Federation Agreements Template.

3.1 Multi-Architecture Systems Engineering Process

Robust, well-defined systems engineering (SE) processes are a key element of any successful development project. In the distributed simulation community, there are several such processes in wide use today, each aligned with

Step	(1) Define Simulation Environment Objectives (2 issues)	(2) Perform Conceptual Analysis (2 issues)	(3) Design Simulation Environment (22 issues)	(4) Develop Simulation Environment (7 issues)	(5) Integrate and Test Simulation Environment (7 issues)	(6) Execute Simulation (1 issue)	(7) Analyze Data And Evaluate Results (1 issue)
Activities	Identify Users/Sponsor Needs **(no issues)** Develop Objectives **(no issues)** Conduct Initial Planning **(2 issues)**	Develop Scenario **(no issues)** Develop Conceptual Model **(no issues)** Develop Simulation Environment Requirements **(2 issues)**	Select Member Applications **(2 issues)** Design Simulation Environment **(14 issues)** Design Member Applications **(1 issue)** Prepared Detailed Plan **(5 issues)**	Develop Simulation Data Exchange Model **(2 issues)** Establish Simulation Environment Agreements **(1 issue)** Implement Member Application Designs **(2 issues)** Implement Simulation Environment Infrastructure **(2 issues)**	Plan Execution **(2 issues)** Integrate Simulation Environment **(1 issue)** Test Simulation Environment **(4 issues)**	Execute Simulation Environment **(1 issue)** Prepare Simulation Environment Outputs **(no issues)**	Analyze Data **(no issues)** Evaluating Feedback Results **(1 issue)**

Figure 1. Multi-Architecture Issues Overlaid on the DSEEP.

a specific simulation architecture such as Distributed Interactive Simulation (DIS), the High Level Architecture (HLA), and the Test and Training Enabling Architecture (TENA). However, there are an increasing number of distributed simulation applications within the Department of Defense (DoD) that require the selection of simulations whose external interfaces are aligned with more than one simulation architecture. This is what is known as a *multi-architecture simulation environment*.

Many technical issues arise when multi-architecture simulation environments are being developed and executed. These issues tend to increase program costs and can increase technical risk and impact schedules if not resolved adequately. One of the barriers to interoperability identified in the LVCAR Final Report [1] was driven by a community-wide recognition that when user communities, aligned with the different simulation architectures, are brought together to develop a multi-architecture distributed simulation environment, the differences in the development processes native to each user community adversely affected the ability to collaborate effectively. Rather than developing an entirely new process, it was recognized that an existing process standard should be leveraged and extended to address multi-architecture concerns. The process framework that was chosen was the Institute of Electrical and Electronics Engineers (IEEE) standard called the Distributed Simulation Engineering and Execution Process (DSEEP).

The LVCAR-I team augmented the major DSEEP steps and activities with the additional tasks that are needed to address the issues that are unique to (or at least exacerbated by) multi-architecture development. These tasks collectively define a "how to" guide for developing and executing multi-architecture simulation environments, based on recognized best practices. Over 40 multi-architecture-related issues were identified, based on an extensive literature search. Each of these issues was aligned with the activity in the DSEEP for which the issue first becomes relevant. This information was provided as an overlay to corresponding information already provided in the DSEEP document for single-architecture development. A tabular representation of the multi-architecture issues aligned with the DSEEP is shown in Figure 1.

The initial prototype of the DSEEP Multi-Architecture Overlay (DMAO) was produced by the team in the summer of 2010, and revised in the winter of 2010-11. The Simulation Interoperability Standards Organization (SISO) has formed a Product Development Group (PDG), in which members of the LVCAR-I team are participating, to take the initial prototype DMAO and evolve it into a consensus-based IEEE standard.

3.2 Federation Agreements Template and Tool

Federation agreements are critical to the successful design, execution, and reuse of federation assets. However,

inconsistent formats and use across federations have made it difficult to capture and compare agreements between federations. This lack of a consistent approach to documenting federation agreements makes reuse and understanding more difficult. Lack of consistent format also prevents tool development and automation. The LVCAR-I team developed a prototype Federation Engineering Agreements Template (FEAT) to provide a standardized format for recording federation agreements to increase their usability and reuse.

The template is an eXtensible Markup Language (XML) schema from which compliant XML-based federation agreement documents can be created. XML was chosen for encoding agreements documents because it is both human- and machine-readable and has wide tool support. Creating the template as an XML schema allows XML-enabled tools to both validate conformant documents, and edit and exchange agreements documents without introducing incompatibilities. Wherever possible, the LVCAR-I team leveraged existing, authoritative schemas for the representation of elements in this schema, including:

- M&S Community of Interest—Discovery Metadata Specification (MSC¬DMS)
- XML Linking Language (XLink)
- XML Metadata Interchange (XMI)
- Common Platform Enumeration (CPE)
- Intelligence Community Information Security Marking (IC-ISM)
- eXtensible Configuration Checklist Description Format (XCCDF)
- Geography Markup Language (GML)

The federation agreements are decomposed into eight categories:

1. Metadata—Information about the federation agreements document itself.
2. Design—Agreements about the basic purpose and design of the federation.
3. Execution—Technical and process agreements affecting execution of the federation.
4. Management—Systems/software engineering and project management agreements.
5. Data—Agreements about structure, values, and semantics of data to be exchanged during federation execution.
6. Infrastructure—Technical agreements about hardware, software, network protocols, and processes for implementing the infrastructure to support federation execution.
7. Modeling—Agreements to be implemented in the member applications that semantically affect the current execution of the federation.
8. Variances—Exceptions to the federation agreements deemed necessary during integration and testing.

The prototype FEAT schema was produced by the LVCAR-I team during 2010. SISO has formed a PDG, in which members of the LVCAR-I team are participating, to take the initial prototype FEAT and evolve it into a consensus-based SISO standard.

Because of the complexity of the schema, the LVCAR-I team recognized that most users would need a tool to be able to implement it effectively. In the winter of 2010¬11, the LVCAR-I team developed an initial prototype FEAT editor tool that implements some key elements of the schema. The tool, which has received an "EAR 99" export designation so that it can be exported to all but a few countries, was provided in its initial prototype form to the SISO PDG so others could experiment with it and improve it. The intent, once required approvals are obtained, is to make the FEAT tool an open-source software product.

Both the prototype DMAO and the prototype FEAT and its editor tool were developed as part of the LVC Common Capabilities technical area of the project. A paper on all of the Common Capability efforts was presented at the 2010 I/ITSEC [3].

4. ADVANCING THE REUSE OF LVC SIMULATION ASSETS

It is generally accepted that LVC simulation assets, like assets in the broader M&S community, have not achieved the desired degree of reuse across DoD. Many reasons for that have been postulated. In attempting to advance the reuse of LVC simulation assets, the LVCAR-I team explored two areas:

- Alternative Business Models for Reuse
- Asset Reuse Mechanisms

4.1 Investigation of Alternative Business Models for Reusing LVC Simulation Assets

The LVCAR Final Report [1], in its Comparative Analysis of the Architectures and Comparative Analysis of Business Models, identified two significant impediments to sharing and reuse of event development tools across programs and communities. The first is the existence of a wide range of tools utilizing a correspondingly wide range of business models. The second impediment is the current environment where different formats are used by the different architectures to store like event data. The LVCAR-I team first undertook a study effort to identify the most beneficial approaches to facilitate tool sharing across architectures based on a structured analysis of the current state.

As a result of the study effort, the LVCAR-I team identified a desired migration path based on the current states of LVC tools, which is shown in Figure 2.

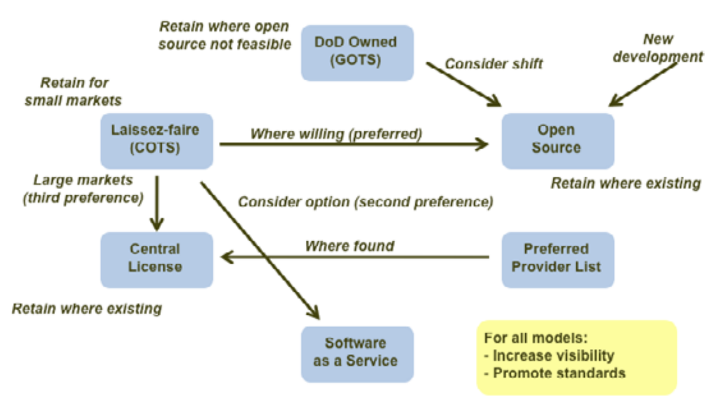

Figure 2. Migration Paths for LVC Tools

Individual long-term recommendations based on the analysis represented in Figure 2 were as follows:

1. For legacy DoD-owned tools, consider a shift to open source, to reduce DoD costs and foster potential innovations. Use the experiences from an open-source pilot to decide if this should be done, and if so, what considerations exist for LVC tools.
2. For new tools, where there is a desire to provide DoD influence but to defray ownership costs, use an open-source model also informed by the open-source pilot.
3. Where small numbers of licenses are purchased from industry, do not make a change.
4. Where a large number of licenses have been and continue to be procured from industry, take the following actions in the order presented until a viable option is identified:
    a. Shift to open source. This assumes that vendors are willing and the open-source pilot experience indicates there is benefit to DoD.
    b. Shift to software-as-a-service. This assumes that vendors are willing and the experiences from a software-as-a-service pilot show benefit to DoD exists.
    c. Attempt to negotiate DoD-wide discounted licenses.
5. For current open-source efforts, make no changes.
6. If preferred-provider lists have been established, attempt to establish DoD-wide discounted licenses, using the experiences gained from a central-licensing pilot.
7. For existing centrally-negotiated licenses, do not make a shift.
8. The study team was unaware of any existing software-as-a-service arrangement for LVC tools, so no recommendations in terms of a shift from current practices are made.
9. For all business models, increase the visibility of what tools are currently used, and take steps to increase the visibility of user experiences as indicated by the LVC Asset Reuse Mechanism effort.
10. Consistent with DoD policy, use open standards as a basis for tool procurements, and participate in standards development activities to ensure DoD's needs are met.

Based on these recommendations, which were published in the summer of 2010, the LVCAR-I team embarked upon attempts to conduct pilot efforts for software-as-a-service, central licensing, and open-source software.

In the software-as-a-service area, a Request for Information (RFI) was drafted, and was issued via a new mechanism provided by the Program Executive Office, Simulation, Training, and Instrumentation (PEO-STRI). A relatively small number of responses were received, which are currently being evaluated for lessons learned for potential future such activities. A specific tool to use for the central-licensing pilot has not yet been identified, but discussions are underway in one area. Finally, for the open-source

pilot, it is likely that the FEAT tool discussed above will be the primary open-source candidate, although sources of existing tools that use traditional business models that might be willing to migrate to an open-source business model are still being investigated.

4.2 LVC Simulation Asset Reuse Mechanisms

As mentioned earlier, the reuse of software, data, and other assets in DoD M&S development is seen as being neither as frequent nor as effective as it could be, and as a consequence, the potential benefits of reuse to the DoD enterprise are not being fully realized. Improvements in the enterprise culture and processes supporting reuse are needed to increase the frequency of reuse. Three alternative approaches to accomplishing those improvements were defined and evaluated by the LVCAR-I team. An assessment of multiple existing repositories using a carefully developed set of M&S¬oriented evaluation criteria was conducted to identify where enhancements are needed.

The LVCAR-I team examined 13 existing M&S catalogs, repositories, and registries of interest to the LVCAR-I effort and evaluated the applicability of these and other reuse initiatives. A detailed model of LVC asset reuse mechanisms based on 22 comprehensive reuse use cases tied to the DoD Net-Centric Data Strategy and commercial standards for repositories was developed and used to facilitate the research and analysis conducted. Consideration of the state of these LVC asset reuse mechanisms, together with feedback from stakeholders within all communities enabled by M&S in the form of questionnaires, workshop discussions, and interaction in the government-industry profession, informed this study and recommendations.

Three complementary approaches to improve LVC Asset Reuse Mechanisms were examined. The Transactional Approach focuses on enhancing the discovery and acquisition of reusable M&S assets through a set of distributed, interconnected M&S catalogs, registries, and repositories. The Social Marketing Approach addresses the long-term improvement of behaviors that promote reuse of M&S assets. The Process-Based Approach encourages more frequent reuse by enhancing reuse guidance within standard DoD M&S systems engineering process models. These three approaches were evaluated in terms of desirability, achievability, and affordability, as well as the likely barriers to their success.

The Transactional Approach was rated as the most affordable due to existing investments and is roughly equivalent to the Process-Based Approach in terms of desirability. The Process-Based Approach was rated as the most easily achievable based on its compatibility with ongoing standards initiatives in M&S systems engineering processes, and also an emerging impetus towards SOAs. A Social Marketing Approach was rated as the least mature in all three indices of desirability, achievability, and affordability, but it offers some unique methods to increase reuse frequency. Barriers to the success of the Social Marketing and Process-Based Approaches were rated as equal in difficulty.

Building upon the results of the initial evaluation, which was completed in the summer of 2010, the LVCAR-I team embarked upon an effort to build a prototype product that would enable better asset reuse. The Enterprise Metacard Builder Resource (EMBR) Portal prototype was completed in early 2011, and is instantiated on a web-accessible server maintained by SimVentions, Inc. It provides the ability to create metacards, based on the MSC-DMS, for LVC assets, allows links to locations where those assets may be obtained, and provides a mechanism for users to comment on their use of the assets, and interact with other users. Further information on the EMBR Portal may be found in Ref. [4].

5. INCREASING THE COMMONALITY OF DATA STORAGE FORMATS

The LVCAR Final Report [1] recommended actions to promote the sharing of tools, data, and information across the DoD enterprise and to foster common formats and policy goals to promote interoperability and the use of common M&S capabilities. One of the recommended actions was to examine different data storage formats used across the various architectures to determine the feasibility of creating a set of architecture-independent formats. Such formats would be used for storage of classes of data in order to mitigate the cost and schedule impacts of database conversion, minimize conversion errors, and improve consistency across LVC architectures. The focus of the LVCAR-I effort in this area is limited to data interchange formats and applicable standards where the data is persistent, e.g., in stored datasets.

The LVCAR-I team identified nine categories of data storage formats, based on expertise and feedback received at the LVC Common Capabilities Workshop at JHU/APL in

November 2009 and questionnaires administered in person at the 2009 I/ITSEC conference and online. This stakeholder feedback was used to assess the priority for rationalization of data storage formats for each category. The team examined the contents of eight metadata standards registries, catalogs and repositories for each category identified. These sources included the DoD Metadata Registry, the DoD Information Technology Standards and Profile Registry (DISR), the North Atlantic Treaty Organization (NATO) and DoD M&S Standards Profile, and the Acquisition Streamlining and Standardization Information System (ASSIST) database, in addition to privately maintained source materials.

For each of the nine format categories, a list of applicable formats was compiled and characterized in terms of currency, openness, maturity, and applicability as a source (producer), interchange (mediation) and executable (consumer) data format. This information was used to assess the difficulty of rationalizing formats within each category.

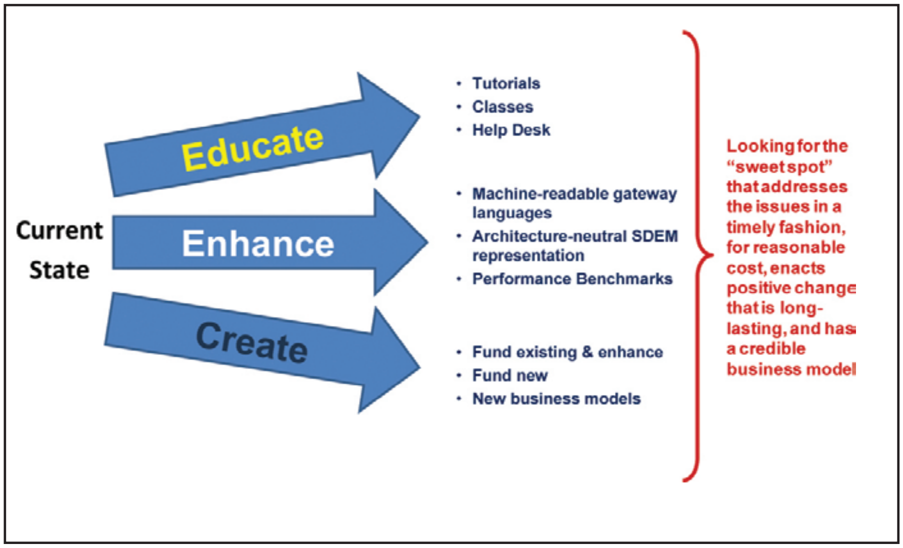

Figure 3. Potential Strategies for Improving Gateways

In addition, the team developed a strategy for each of the nine categories by evaluating the feasibility of moving to a state of greater reuse via a combination of:

1. Reduction in the number of formats used in each category;
2. Standardization of formats in each category if no standards exist;
3. Increased adoption of mediation formats to reduce translation errors; and
4. Creation or engagement with category-specific communities of interest (COIs).

Using this prioritization approach, the team concluded that the standardized formats should be pursued in the following order:

Priority 1.
1: Manmade features and event results Priority
2: Geospatial Priority
3: Unit Order of Battle (UOB) and Plans

Priority 2. scenarios.
4: Platform/weapons performance and behavior
5: Electronic Order of Battle (EOB)/network and

Priority 3. logistics.

The initial assessment effort was completed in the early summer of 2010. Based on an assessment of where these priorities were already being investigated or planned to be investigated within the broader DoD community, as well as the expected cost of developing reasonable solutions, the team narrowed its focus to making specific recommendations in five of the original nine categories starting in the summer/fall of 2010:

1. Three-dimensional (3D) manmade features
2. Platform/weapon performance/characteristics
3. Event results
4. Logistics
5. UOB / force structure

Of the above, there appeared to be little Service-based interest in standardization of platform/weapon performance/characteristics. In the logistics area, extensions to the Joint Land Component Constructive Training Capability (JLCCTC) Entity Resolution Federation (ERF) logistics data model and representations were recommended.

In the 3D manmade features category, the LVCAR-I team has developed recommendations for specific extensions to X3D. In the event results category, although mature after-action review systems exist, the data formats they use are

both non-standardized and custom-tailored to the data model of the simulation they are logging. So the LVCAR-I team has developed a draft XML schema for event results. In the UOB / force structure category, the team continues to monitor the progress in standardized data formats being performed by Military Scenario Definition Language (MSDL) and Coalition Battle Management Language (C-BML) efforts. Technical papers on the team's work in each of these areas are being prepared for presentation at the 2011 Fall SIW.

6. IMPROVING THE USE OF GATEWAYS AND BRIDGES FOR LVC SIMULATIONS

The LVCAR Final Report [1] presented a vision for achieving significant interoperability improvements in LVC simulation environments. The study recommended several activities intended to reduce the time and cost required to integrate mixed-architecture events. Three of the key LVCAR report recommendations were to determine whether existing gateway and bridge applications were effective in meeting user requirements, whether improvements in gateway/bridge capabilities were necessary to address identified gaps, and how these improvements could be best implemented to maximize DoD return on investment (ROI). The term "bridge" in this context refers to intelligent translators that link together enclaves of simulations that use the same underlying simulation architecture. A "gateway" is also an intelligent translator but is designed to link simulation enclaves that use dissimilar architectures.

Early during the LVCAR-I effort, the team performed gateway and bridge literature research, compiled the team's gateway and bridge usage and development experience, and developed formal gateway and bridge operation terminology. At this point, it became clear that the distinction between "gateway" and "bridge" was moot from a development and usage standpoint. Starting with the initial delineation of capabilities, the team compiled a Gateway Capabilities Matrix Template, and created two structured questionnaires (one for gateway developers, and one for gateway users). A one-day workshop, the "LVCAR Common Gateways and Bridges Workshop," was held in March 2010 to present the findings of the questionnaires. There was wide agreement that there are several potential improvements that can be made that will lower the technical and cost risks generally associated with the use of gateways.

Based on the above, in the early summer of 2010, the team developed three potential strategies for execution, referred to as Educate, Enhance, and Create (as well as a "Status Quo" strategy), as shown in Figure 3. Of the four strategies, the team recommended that the Enhance strategy be executed, because it was perceived to have the greatest ROI. More information on the LVCAR-I team's first year of efforts on gateways may be found in Ref. [5].

The LVCAR-I team then embarked on the execution of the Enhance strategy, also incorporating the tutorial recommendation given as part of the Educate strategy. During the past year, the LVCAR-I team has developed the following products:

- A *Gateway Configuration Model* that identifies an explicit set of gateway requirements, and discusses how the emerging gateway products and processes will address those requirements.
- A *Gateways Capability Description* document, which formally delineates the various capabilities that individual gateways can offer to user programs, along with specific levels of implementation for each unique capability.
- An assessment of the *Architecture-Neutral Data Exchange Model* (ANDEM), originally developed by the Joint Composable Object Model (JCOM) effort, to support Simulation Data Exchange Model (SDEM) mapping and/or translation in gateways.
- A preliminary set of *Gateway Performance Benchmarks* (GPBs) to identify specific gateway performance measures, along with use cases that describe how and where these measures should be applied.

The following efforts are in progress:

5. Development of a common *Gateway Description Language* (GDL), in a machine-readable format/syntax, for describing both user gateway requirements and the capabilities that individual gateways can offer, to support user discovery of needed gateway capabilities.

- Development of a common *SDEM Mapping Language (SML)* to formalize format and syntax of mappings between different SDEMs, to reduce the number of required mappings, and to support reuse of mapping data.
- Development of a repository for GDL-based gateway descriptions, incorporating applicable search and requirements-to-capabilities matching algorithms.
- Development of tools for GDL and SML file creation and editing.

- Development of SML Translators for selected gateways
    - JBUS and Gateway Builder (GWB) are likely choices
- Socialization of the preliminary set of GPBs with gateway developer organizations, incorporation of feedback, and preparation of a formal specification.
- Development of a gateways tutorial.

Early work in the second year of effort on gateways is documented in Refs. [6] through [8].

7. LVC ARCHITECTURE CONVERGENCE – PERHAPS A BRIDGE TOO FAR

The LVCAR Final Report [1] developed a vision for achieving significant interoperability improvements in LVC simulation environments. The study recommended activities proposed to lower the time and cost required to integrate mixed architecture events by building better bridges between the legacy architectures (discussed in the previous section) and making the architectures more compatible. As part of the LVCAR-I effort, the team explored converging the current architectures.

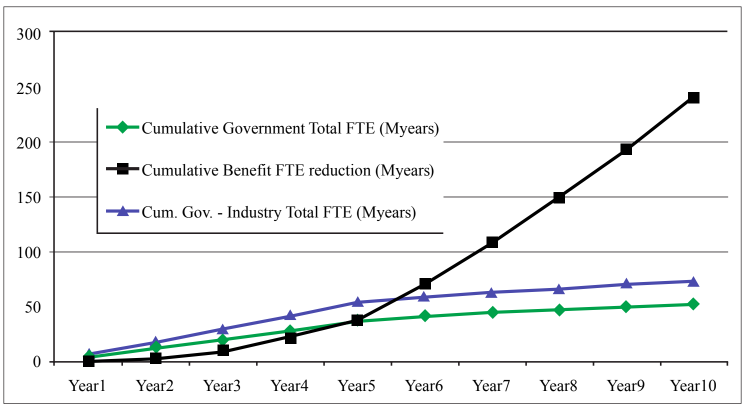

Figure 4. ROI Estimates for Architecture Convergence

Rather than, for example, making the current HLA like the current TENA, the team's goal was to make future HLAs more like future TENAs. Subject matter experts (SMEs) from each architecture (HLA, TENA, DIS, and the Common Training Instrumentation Architecture (CTIA), participated together on the LVCAR-CT. Each SME provided existing documentation resources and identified where in the documents to extract the key services and tools. Using this information, the team first developed a document that characterized the existing architectures.

The next step was to determine what actions would lead to convergence. The vision was that in 2015, new versions of CTIA, DIS, HLA, and TENA would emerge that would incorporate the results of the convergence effort. The LVCAR-I team described the envisioned converged architecture in terms of how it would execute in a multi-architecture event. This converged execution would contain

1. Simulations that need not be aware that multiple architectures are in use,
2. Parts of the support infrastructure of the legacy infrastructures, and
3. A common shared library for communication.

This concept was selected because it requires no changes to the simulations (which are the area of greatest DoD M&S investment). As a result, changes under this proposed solution would impact only a few infrastructure providers and require significantly less investment to achieve convergence. Construction of software to gradually evolve legacy infrastructures and achieve convergence would involve several years of effort.

As part of its first-year efforts, the LVCAR-I team also calculated an estimate of the investment that would be required over a number of years to achieve the envisioned converged state, as well as the return that was expected to be realized over many years. Those estimates are shown in Figure 4. Upon review of the timelines and costs, the government decided, because of the magnitude of the investment, and the number of years required to achieve a "break-even" state, to terminate any continuing effort on architecture convergence activities during the summer of 2010. More details on the year-long convergence effort may be found in Ref. [9].

8. INVESTIGATING THE APPLICATION OF ADDITIONAL TECHNOLOGIES TO LVC SIMULATIONS

In addition to the primary areas of investigation discussed above, the LVCAR-I team was asked to look toward the future at different technologies that might improve LVC simulations. The first, SOAs, have been in use in other communities, so the question is the degree to which SOA might apply to LVC simulations. To address this,

a study of benefits and barriers of SOA application to LVC was undertaken by JHU/APL members of the LVCAR-I team, and a pilot application of SOA to LVC was undertaken by MITRE. Looking even farther ahead, members of the JHU/APL LVCAR-I team undertook a small "LVC Futures" study to see how future technologies might be applied to LVC simulations in the 2025 timeframe.

8.1 Service-Oriented Architectures – Benefits and Barriers

The goal of the DoD to reuse M&S assets, such as visualization software, data management applications, and interoperability middleware, is similar to the goal of the business community to reuse existing business applications in the architectural paradigm of SOA. This common association with reusability suggests that the integration of SOA and distributed simulation would be a good combination, but not all M&S applications lend themselves to SOA-based solutions.

Although SOA infrastructures can and have been applied to mid-size LVC distributed simulations in a few efforts, the question remains, is SOA a good fit? To answer this question, an examination of the benefits of and barriers to the application of SOA to LVC distributed simulations is required. The LVCAR-I team enumerated eight benefits of, and seven barriers to, applying SOA to LVC distributed simulations.

In short, SOA was assessed to be an excellent architectural choice for an LVC distributed simulation if the criteria below are met.

- If there exist multiple contributors to a LVC distributed simulation that have clearly defined areas of interest, each have a willingness to take ownership, and all have a stake in the overall success of the resulting simulation system.
- If there is a critical need for the ability to dynamically add components, allow updates to keep components current, or reconfigure a system in relatively short order.
- If the simulation components are well-encapsulated through the use of an agreed upon common SDEM for the LVC distributed simulation.
- If all simulation components will be operating at similar levels of abstraction for the objects and interactions within the simulation.
- If all simulation components will be operating at the same echelon of security.
- If the modeling, visualization, and management control can be segmented within the infrastructure.
- If translation components, i.e., gateways, can be incorporated into the federations, or are definable as services themselves, then scalability of the system is increased.
- If a business model can be defined and maintained where it is beneficial to share the cost of LVC distributed simulations.

On the other hand, SOA was assessed to be a poor architectural choice for an LVC distributed simulation if any of the following conditions exist.

- If all parties cannot agree on goals, interfaces, and an evolution plan, and the ability to record these agreements in governance documents.
- If the funding and time are not available to permit components to be written so that they are usable in a more general way, are available as a service, and external requests for updates are heeded.
- If the LVC distributed system being developed does not need to be updated frequently to meet its goals, such as static training, testing, experimentation, or demonstration that is unchanging; then, SOA is too heavyweight an infrastructure.
- If throughput is a significant concern, as SOA infrastructures are traditionally written as remote services employing request-response-based communication protocols, such as web services.
- Being able to create services out of simulation components is difficult, since simulation components are not traditionally a request-response entity. However, the SOA infrastructure most likely should not be applied at the level of the simulation component, but rather at the level of the simulation cluster. The exception to this is that the SOA infrastructure can be used exclusively to initialize, configure, and deploy simulation components, and allow the distributed simulation infrastructures, i.e., HLA, TENA, and DIS, to process the simulation-to-simulation-component communication.
- Current simulation infrastructures are often composed in brittle ways. If components are reconfigured and redeployed on-the-fly, the distributed simulation is not likely to not operate properly. Most components would have to be updated to handle the challenges of rapid deployment.
- The use of a SOA infrastructure in the DoD M&S community is only cost effective if the system gains a wide-enough acceptance for the services to be used.

In summary, SOA does appear to have a greater upfront cost and may provide a greater cost savings over the long-term through reuse and a potentially cost-effective business model, such as software-as-a-service. SOA requires greater cooperation among distributed simulation developers than traditional development. In addition, the challenges associated with SOA are both political and social, as well as technical. Whether the successes achieved on a few mid-size distributed simulation tasks can be scaled up to full-size simulation exercises still remains to be seen.

In addition to documenting the benefits and barriers of SOA application to LVC simulations in a technical report, the LVCAR-I team has produced a tutorial on this topic. Evolving versions of the tutorial were presented at the 2010 Fall SIW, the 2010 post-I/ITSEC tutorial session, and the 2011 Spring SIW. A DVD of the tutorial has also been produced and distributed to the LVCAR-I DoD sponsoring organizations, to enable the tutorial's use in a non-classroom setting.

8.2 Service-Oriented Architecture Pilot Effort

The concept of using SOA-based software technologies is not new and is being eyed with keen interest by many in the simulation industry. However, no extant program of record can afford to put their program at risk on an unproven approach, no matter how promising. The 2008 DoD study, LVCAR Final Report [1], recommends to "Take actions that can reduce or eliminate the barriers to interoperability across the architectures." As an early step toward addressing the LVCAR recommendation, a "SOA Outlook" pilot effort was developed to determine if commercial SOA

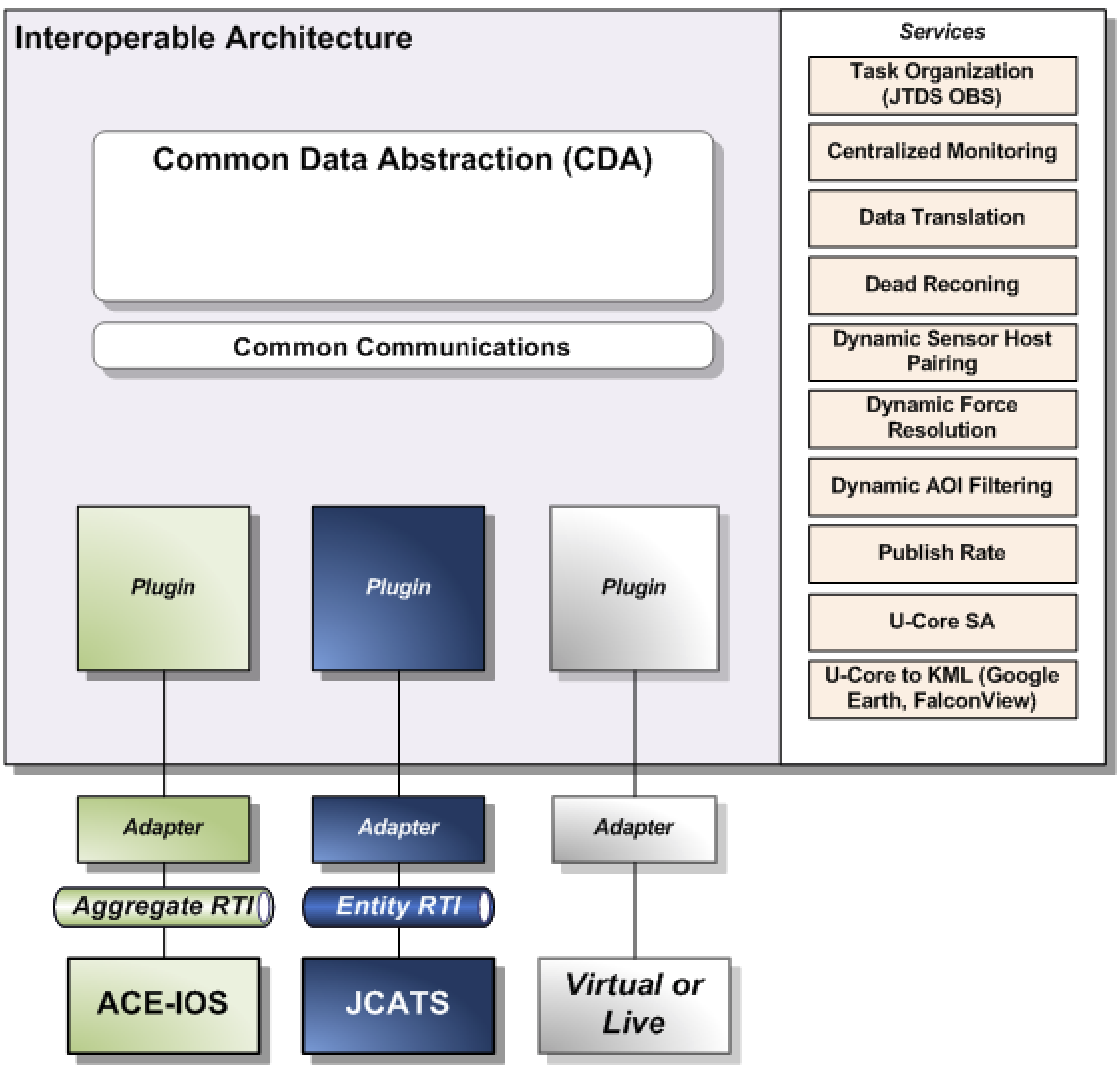

Figure 5. LVC SOA Pilot Architecture

architectures, software, and principles are an appropriate solution space for achieving LVC interoperability.

The pilot was designed around the use of open standards wherever possible and attempts to illustrate SOA principles like composition and reuse. A common data abstraction layer in the application server provided an abstraction of the storage mechanism through the Java Persistence Application Programming Interface (JPA) standard and allowed for non-system-specific storage of shared data. Integration with existing legacy systems used a two-part adaptor / plug-in architecture where the adaptor connects directly to the existing infrastructure and communicates with its plug-in counterpart inside the application server infrastructure. (See Figure 5.) The pilot also included a sample of other services that would be required for a complete interoperability framework.

The SOA pilot successfully provided a limited interoperability framework based on the constraints of the use case selected and the level of effort involved. Cursory performance data was also gathered and reported.

8.3 Potential Future Technology Application to LVC Simulation

The "LVC Futures" study effort set out in 2010 to investigate emerging technologies and processes in the 2025 timeframe and their impact on M&S activities in support of future DoD activities. By first proposing a set of possible future operational vignettes (e.g., military, disaster relief), the LVCAR-I team applied near-term technologies that could have substantial impact in an M&S context for the DoD and other coalition partners within the context of these vignettes. Additionally, the LVC Futures task looked towards processes that would impact future socialization of and collaboration within M&S.

To frame the technology investigation, the team generated five vignettes to capture the scope of future operational needs. Each of these vignettes included consideration of:

- Operations – conventional, cyber, joint, stability/aid, irregular, counter-insurgency
- Services – Army, Navy, Air Force, Marine Corps
- Reserves / National Guard
- Time Horizon – weeks, months, years
- Foreign military participation
- Non-governmental organizations / Corporations

Using the vignettes, the team estimated a technology's impact for M&S activities, including skill training, unit training, mission planning, environmental analysis, C4I structure, and acquisition. These impacts were summarized in tables for each technology to create an effects matrix with seven possible gradations of impact.

Technologies and processes were binned into nine categories in the broader areas of implementation, and socialization and adaptation, as follows.

Implementation

- Mobile computing and augmented reality
- Ubiquitous surveillance and automated reasoning
- Event-model driven architectures
- Self-healing / self-managing systems
- M&S social graph Socialization and adaptation
- Crowd-sourcing
- Mash-up software and FIST (Fast, Inexpensive, Simple, Tiny)
- Cloud encapsulation
- Everything is a game

Results of the team's efforts during 2010 are given in Ref. [10]. In the summer of 2011, an implementation plan is being prepared for a potential prototype for rapid situational awareness that builds on the "everything is a game" category above.

9. THE WAY AHEAD

The LVCAR-I task was approved for continuation through fiscal years 2011 and 2012 by the DoD M&S Steering Committee. The LVCAR-I team is currently building upon the accomplishments in the first two years described in this paper to advance LVC processes and products.

In the standards area, working through SISO in conjunction with the larger M&S community, the DMAO is expected to become an IEEE standard, and the FEAT a SISO standard, by the end of the LVCAR-I project. Similarly, the tool to aid users in implementing the FEAT is expected to become a complete product, under an open-source licensing arrangement.

Lessons learned in the exploration of alternative business models for DoD LVC tools, including the use of open source software, software-as-a-service, and central licensing, will

be documented so that future LVC tool developments can take better advantage of these business models. Common data storage format advances in several areas will have been made by the end of the LVCAR-I project in the areas of 3D formats and battle management languages, which will provide a strong baseline for future efforts in this area by other projects, such as the Rapid Data Generation (RDG) effort.

Gateway users will have automated tools at their disposal to aid in discovering appropriate gateways for specific uses. Additionally, some common components for SDEM translation will have been developed to aid in the application of gateways. Building on the EMBR portal, an LVC asset reuse repository will be available to support LVC gateway discovery and reuse, which can serve as a model for expanded repository efforts for the broader M&S community.

ACKNOWLEDGMENTS

The authors would like to recognize the technical contributions of all of the LVCAR-I team members, whose work is described in this paper. The majority of those individuals are listed as authors/co-authors of the technical papers listed as references below.

REFERENCES

[1] Live Virtual Constructive Architecture Roadmap (LVCAR) Final Report (2008).

[2] Coolahan, J. E., Morse, K. L., Saunders, R., Lutz, R. R., Riggs, W. C., et al, "Live-Virtual-Constructive (LVC) Architecture Roadmap Implementation Workshop," in *Proc., 2010 Spring Simulation Interoperability Workshop*, Orlando, FL (Apr 2010).

[3] Morse, K. L., Brunton, R. P., Riggs, W. C., and Lutz, R. R., "Live Virtual Constructive (LVC) Common Capabilities," in *Proc., 2010 Interservice/Industry Training, Simulation & Education Conference,* Orlando, FL (Nov-Dec 2010).

[4] Riggs, W.C., Morse, K. L., Vick, S. D., Brunton, R. P., Schloman, J. F., Gustavson, P., Rutherford, H,, Chase, T., and Belcher, J., "Emerging Solutions for LVC Asset Reuse," in *Proc., 2011 Spring Simulation Interoperability Workshop,* Boston, MA (Apr 2011).

[5] Drake, D. L., Lutz, R. R., Cutts, D. E., Lessmann, K., and O'Connor, M. J., "LVC Common Gateways and Bridges," in *Proc., 2010 Interservice/Industry Training, Simulation & Education Conference*, Orlando, FL (Nov-Dec 2010).

[6] Lutz, R. R., and Drake, D. L., "Gateway Concepts for Enhanced LVC Interoperability," in *Proc., 2011 Spring Simulation Interoperability Workshop,* Boston, MA (Apr 2011).

[7] Lessmann, K., Cutts, D. E., Horner, N. C. and Drake, D. L., "LVCAR Enhancements for Selecting Gateways," in *Proc., 2011 Spring Simulation Interoperability Workshop*, Boston, MA (Apr 2011).

[8] O'Connor, M. J., Lessmann, K., and Drake, D. L., "LVCAR Enhancements for Using Gateways," in *Proc., 2011 Spring Simulation Interoperability Workshop,* Boston, MA (Apr 2011).

[9] Saunders, R, "Live Virtual Constructive Architecture Roadmap (LVCAR) Convergence Approaches," in *Proc., 2010 Interservice/Industry Training, Simulation & Education Conference*, Orlando, FL (Nov-Dec 2010).

[10] Schloman, J. F., and Labin, J. W., "Future Technologies and Processes and their Impact in the Domain of Live-Virtual-Constructive Architectures," in *Proc., 2011 Spring Simulation Interoperability Workshop,* Boston, MA (Apr 2011).

This article is reprinted from the Fall 2011 Simulation Interoperability Standards Organization (SISO) Simulation Interoperability Workshop (SIW). Paper 11F-SIW-020.

HIGHLIGHTS OF THE CREATE PROGRAM

Douglass E. Post
for the CREATE Team[a]
Department of Defense High Performance Computing Modernization Program,
Lorton, VA
post@hpcmo.hpc.mil

ABSTRACT

The Computational Research and Engineering Acquisition Tools and Environments (CREATE) Program was established as a new 12-year program in FY2008 by the DoD. The CREATE goal is to enable major improvements in DoD acquisition engineering design and analysis processes by developing and deploying scalable, multi-disciplinary, physics-based computational engineering software products for the design and analysis of DoD Ships, Air Vehicles and Radio Frequency Antennas. Meshing and Geometry generation is being provided by a fourth project, MG. CREATE is a multi-institutional, multi-service, multi-agency and multi-disciplinary program with participation by the Navy, Air Force, Army, Office of the Secretary of Defense, industry and academia. The CREATE products are being developed and released on an annual cycle. In 2010, the program released five new products: SENTRI 1.0--RF antenna design; NESM 0.1--Ship Shock Analysis; IHDE 1.0—Ship Hydrodynamic Design and Analysis; Kestrel 1.0—Fixed wing air vehicle analysis; and Helios 1.0—Rotorcraft analysis. Enhanced versions of these products will be released every year starting in 2011. In 2011, five additional products will begin annual releases: DaVinci—a tool for the rapid physics-based design of air vehicles; RDI— an integrated suite of tools to enable rapid physics-based design of naval ships; Firebolt—components to provide models for gas turbine propulsion systems for Kestrel and Helios; NavyFoam—a high fidelity hydrodynamics analysis tool for predicting drag and resistance, seakeeping and seaway loads; and Capstone—components to enable the generation of geometries and meshes for all of the other products. The CREATE products are designed to be modular, maintainable, extensible, and scalable. To accomplish this, the CREATE team has developed a set of software engineering and software project management practices and processes that strike the appropriate balance between the agility and flexibility, and organizational structures and planning that are appropriate for developing complex physics-based, scalable and sustainable engineering software.

ACKNOWLEDGMENTS

The authors are grateful to ASD R&E, the HPCMP Director and the HPCMP office, and the Navy, Air Force and Army for support and encouragement.

INTRODUCTION

A continuing challenge at every level of the DoD acquisition community has been to significantly reduce acquisition time and attendant acquisition cost. Many DoD acquisition programs are behind schedule, exhibit cost overruns, and fail to meet requirements[1]. Every Secretary of Defense in last twenty or more years has made acquisition improvement a major priority. As one step to improve the performance of some of these programs, the DoD established the Computational Research and Engineering Acquisition Tools and Environments (CREATE) Program in 2008. CREATE is a 12-year Program to develop and deploy three computational engineering tool sets for the engineering organizations (government and industry) who support the DoD acquisition programs. 70% of its support comes from the Office of the Secretary of Defense and 30% support from the relevant service organizations. CREATE focuses on improving the quality of engineering design and analysis in three major technical areas: Air Vehicles, Ships and Radio Frequency Antennas. The approach is to use multidisciplinary physics-based design tools early in the acquisition process to develop higher quality designs that are better optimized and more mature and have fewer design flaws so that they require less rework that result in delays and increased cost.

A key feature of the DoD acquisition programs is the need for extensive design and testing of model and full-scale prototypes. These include wind tunnel models and

[a] The CREATE Team consists of approximately 75 engineers and scientists from the Office of the Secretary of Defense, Navy, Air Force, and Army and industrial support contractors.

full scale flight test aircraft. Products have been developed this way since the beginning of the industrial revolution. However, repeated design-build-tests of these physical prototypes take lots of time and money and is proving to be unsustainable for industry and government. By facilitating a "Design Through Analysis" paradigm, CREATE will enable engineers to quickly create and analyze "virtual" prototypes. This substantially reduces the number of required physical prototypes. In addition, while physical prototypes are usually employed only after Milestones B and C, virtual prototypes can be utilized early in concept development well before Milestone A. Industry is beginning to make extensive use of physics-based computational design and analysis to reduce the need for physical prototypes[2]. Goodyear Tire was able to reduce their time to market from 3 years to less than one year[3]. This greatly improved their competitive position. The US Nuclear weapons program has employed virtual prototypes for the development of the nuclear stockpile since the 1950s[4]. In the past, nuclear weapon tests were costly, hazardous, unpopular, and hard to diagnose. Now, there are no nuclear weapons tests. "Design through analysis" supplemented by the testing of small scale components is the only option for maintaining the nuclear stockpile.

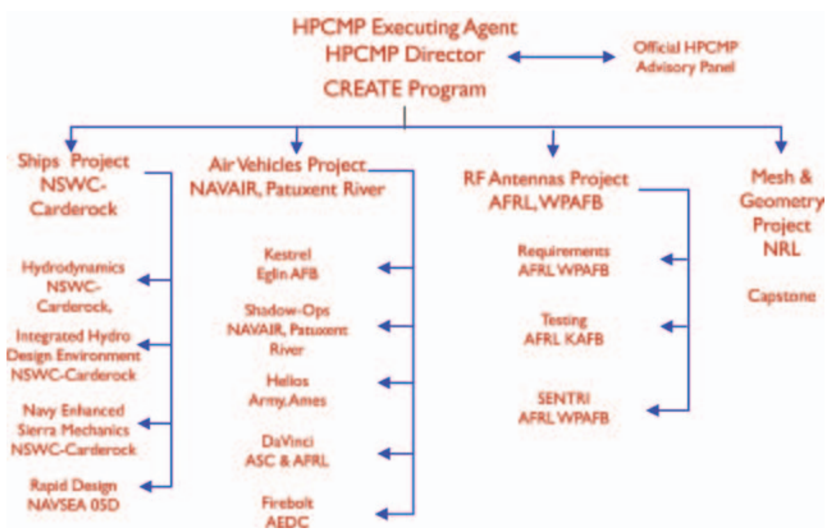

Figure 1. Three-Service CREATE Program Structure

While the goal of the CREATE program is to "Enable major improvements in DoD acquisition engineering design and analysis processes by developing and deploying scalable physics-based computational engineering software products," it has many complementary objectives:

- Reduce reliance on empirical design by the use of physics-based computational design validated by experimental testing
- Detect and fix design flaws early in the design
- Increase design option space
- Promote early system integration
- Enhance ability to respond to rapidly changing requirements
- Enhance engineering workforce productivity
- Enhance the ability of the DoD to develop and deploy physics-based computational engineering software

The enabling technology for "Design Through Analysis" is the combination of the exponential growth in computational power over the last 65 years from one Floating Point Operation/Second (FLOPs) to 10^{15} FLOPs and application software that can utilize these systems[5]. Today, 10^{15} FLOPs supercomputers are available at only a few institutions worldwide but within five to ten years, this level of computer power will be widely available. Already, one can purchase special purpose 10^{12} to 10^{13} FLOPs computers for ~ $5k and affordable general purpose supercomputers are not far behind. The improvement in speed has been so rapid that generally one only has to wait ~ seven years for the speed of the 500th highest speed supercomputer to match the speed of the 1st ranked supercomputer. With such computers, design engineers will be able to run applications that can provide accurate and timely predictions of the performance of complete weapons systems by including all the important effects that determine system performance. The CREATE program is producing four sets of scalable software tools to enable the engineering organizations that support the DoD acquisition programs for Air Vehicles, Ships and Radio Frequency antennas to improve the design process by enalbling those programs to exploit this computer power.

CREATE PROGRAM OVERVIEW

CREATE is a multi-disciplinary, multi-institutional program. It is led by the High Performance Computing Modernization Program in the Office of the Director, Defense and Engineering. The CREATE Air Vehicle project is developing: software tools for high-fidelity, full-vehicle, multi-physics detailed design and analysis of both fixed wing aircraft (Kestrel) and rotorcraft (Helios); components for the fixed wing and rotorcraft propulsion effects (Firebolt); and a tool for rapid, early stage design (DaVinci). The CREATE Ship design project is developing: software applications to analyze and predict the shock and damage effects and reduce the need for tests to assess ship shock and damage effects due to underwater and air explosions (Navy Enhanced Sierra Mechanics--NESM); software tools to facilitate use of the existing Navy design tools including easy access to the Navy's product model database of the present Naval Fleet (Integrated Hydro Design Environment--IHDE); software to provide high fidelity analysis and prediction of the hydrodynamic performance of ships for resistance and drag, seaway loads and seakeeping (NavyFoam); and software tools for rapid development, assessment, and integration of candidate ship designs to avoid cost versus capability mismatches(Rapid Design Integration, RDI). RDI is a partnership of CREATE with the Office of Naval Research (ONR) and NAVSEA. The CREATE Radio Frequency(RF) Antenna project is developing the SENTRI tool for both detailed and rapid design and analysis for RF antenna designs and their integration with land, sea, air and space platforms. A key focus is computational algorithms that improve scaling from n^3 to $n \cdot \log n$ to enable the analysis of large, complex antenna arrays on large platforms ($n \sim 10\ l/\lambda$) where l is platform linear dimension and λ is the radar wavelength. Present methods for generating the geometries and meshes that are the starting point for analysis can take weeks to months, and are a major bottleneck. A fourth project (Meshing and Geometry—MG) has been formed to improve the ease, speed, flexibility and quality of geometry and mesh generation through development of a set of components (Capstone) for the three major projects.

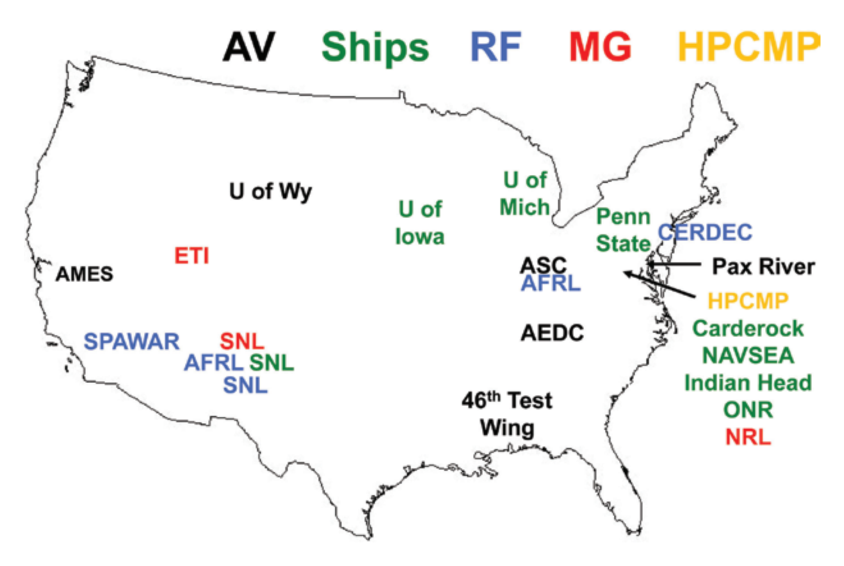

Figure 2 CREATE Project and Product Team locations

The projects concentrate on improving the physics- and mathematics-based engineering design and analysis capability including aerodynamics, structural mechanics, propulsion, hydrodynamics, control, shock and vibration damage, electromagnetics, and geometry and mesh generation and analysis. The CREATE tools are being designed to support the entire acquisition life cycle from rapid conceptual design through detailed design to sustainment and refurbishment.

The CREATE program is being executed by a set of non-collocated government-led teams of government employees and contractors from the Air Force, Navy and Army (Figures 1 and 2). While this approach leads to considerable challenges for program coordination and management, it has helped the program attract the necessary highly-skilled teams with the requisite subject-matter expertise that are embedded in their customer organizations.

The CREATE program has developed a set of Software Engineering Practices and Processes to ensure the development of software products that are scalable, maintainable, extensible, portable and reliable software (Table 1). These 66 practices are designed to balance agility and flexibility with short and long term planning. Each software product has a technical description, and developer and user manuals. The CREATE projects are emphasizing Verification and Validation plans and tests. The projects are implementing Earned Value Management (EVM) processes to measure their progress and improve their estimation and planning processes. Agile development methods such as SCRUM are

being adopted by most of the development teams[6]. Each development team has adopted an annual release cycle for developing and releasing products (Figure 3). In a given year, each team is completing last year's release, developing this year's release, and planning next year's release. The systems engineering approach involves three major steps: 1) Develop and implement methods for integrating the different physics elements of each multi-physics applications (e.g. computational fluid dynamics + structural mechanics + control + propulsion + six-degrees-of-freedom +… for aircraft); 2) Develop and implement methods for to improve the scaling for the application software to fully exploit the available computational power; and 3) Sustain the software for the acquisition community. The annual release cycle ensures that the CREATE projects gets feedback on the utility of their products as early as possible. This allows them to improve the usability of the software, to track and incorporate evolving requirements from the user communities, and to take advantage of emerging opportunities to improve the quality and capability of the software and to exploit unforeseen "market opportunities" as they emerge during the life of CREATE while staying on track to meet the program's original goals.

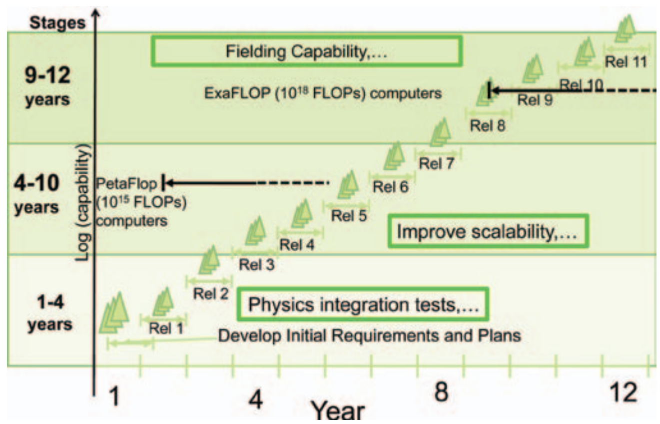

Figure 3 CREATE Systems Engineering Approach and Annual Release Strategy

CREATE PROGRESS IN 2010

Five CREATE products had beta releases in FY2009. The CREATE Ships project released IDHE 1.0 and NESM 0.1. The project also continued the development and assessment of a high fidelity hydrodynamics code for accurate prediction of hydrodynamic resistance and drag, and requirements for seakeeping and seaway loads (NavyFoam). In partnership with Naval Sea Systems Command (NAVSEA) and the Office of Naval Research (ONR), the Ships Project established a Rapid Design Integration product group (RDI) to quickly develop conceptual designs for new ships and assess modifications of the ships in the existing fleet.

The Integrated Hydro Design Environment provides a user friendly interface for the Naval design community (naval architects and marine engineers) to utilize many of the different hydrodynamic design tools that the Navy has developed over the last 40 years[7]. Using IDHE 1.0, users can retrieve product models for existing ships in fleet from the LEAPS product model database, prepare the input to run any of the candidate design tools, run the tools and analyze the results (Figure 4). Without IHDE, users would need to become expert in dozens of different analysis codes and also learn how to utilize LEAPS. This has proved to be a significant barrier to the use of computational design tools by the Navy and their contractors. IHDE is already being utilized to train naval architecture interns at the Naval Surface Warfare Center (NSWC) at Carderock, and by engineers at Carderock and other centers.

The CREATE Ship Shock product team has the goal of assessing and predicting the vulnerability of naval vessels to explosions below the water and in the air. For underwater explosions, the major calculation elements are calculation of 1) the underwater detonation and the transmission of the detonation pressure wave through the water, 2) coupling of the pressure wave to the ship hull, 3) transmission of the shock through the ship structure, and 4) assessment of the shock conditions in the ship compartments and structure for equipment damage. The Dysmas code[8], jointly developed by the US and Germany, is presently used for this analysis. The code is being modified and portions replaced to improve the performance and accuracy. In particular, the structural mechanics package doesn't scale well, and is being replaced by a portion of the Sierra Mechanics Suite developed by the Sandia National Laboratory that has very good scaling properties. The goal is to develop a predictive capability for shock vulnerability quickly enough for use in developing ship designs that are less vulnerable to damage instead of the present practice of assessing the vulnerability

Software Engineering Practices

1. Requirements Management and Stakeholder Engagement
2. Software Quality Attributes
3. Design and Implementation
4. Software Configuration Management
5. Verification and Validation of CREATE Products
6. Software Release
7. Customer Support

Table 1 Key Software Engineering Practice, Document and Plan Categories

after the ship is built. The partnership with Sandia is proving to be a great asset for CREATE. Sandia is developing an excellent set of scalable engineering codes, and the DoD can use some of these codes to solve hard problems (both physics and software scalability aspects) without spending the time and resources to duplicate the capability that Sandia has already developed as part of its DOE funded mission. In addition, CREATE has adopted a number of good software development practices that Sandia introduced into our program, such as SCRUM.

The Ship shock product team has stated their requirements in terms of six use cases (Table 2). This has proven to be a highly effective way of capturing the relevant requirements, translating them to requirements of code development and planning, and communicating the planned product capabilities to sponsors and users. The first release is named NESM 0.1 (Navy Enhanced Sierra Mechanics) because it provides about 90% of the capability required for a full analysis of Use Case I. NESM 0.1 has been benchmarked with the existing shock analysis tools, and provides answers as good or better than the existing tools. The agreement with validation data is also good. NESM 1.0, planned for release in late FY10, provide all of the capability required for Use Case I and part of the capability required for Use Case II.

The CREATE Radio Frequency Project released SENTRI 1.0 for beta testing in late CY09. SENTRI 1.0 delivered an initial capability for modeling antennas, periodic structures, and microwave circuits and predicting radar cross sections. SENTRI 1.0 has the capability to model antenna performance for simple antennas (patch, notch, horn, and spirals) for radar, communication and GPS antennas, simple phase array antennas, and cavity backed antennas. It can handle periodic structures including frequency selective surfaces, circuit analog absorbers, meta-materials, and infrared filter and absorbers.

It can also calculate power splitting, and predict the effects of materials and the performance of filters and circulators for microwave circuits. Its capability is comparable to the existing commercial computational electromagnetic packages.

The focus on SENTRI 1.0 was to integrate a collection of solution methods to achieve the required SENTRI functionality. SENTRI 1.0 is based on the use of Finite Element-Boundary Integral Methods chosen for accuracy, reduced volume meshing and adaptability. This resulted in order n^3 complexity and order n^3 scaling. As noted earlier, $n \sim 10\, l/\lambda$ where l is platform or antenna linear dimension and λ is the radar wavelength. A large part of the SENTRI 1.0 development effort involved rapid assembly of algorithms, packages and techniques from myriad sources. This resulted in a code that did provide the initially required capability to test integration and scaling issues and to get feedback from the user community, but was accomplished without the

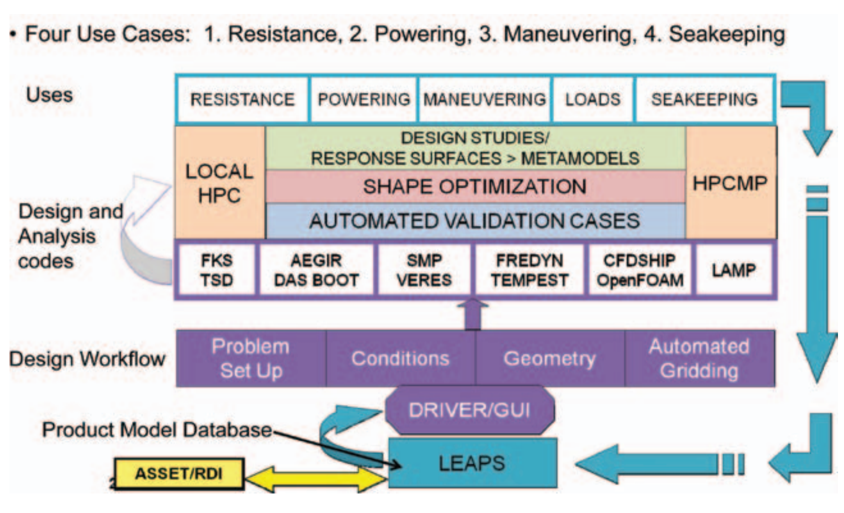

Figure 4 Integrated Hydro Design Environment (IHDE) Architecture

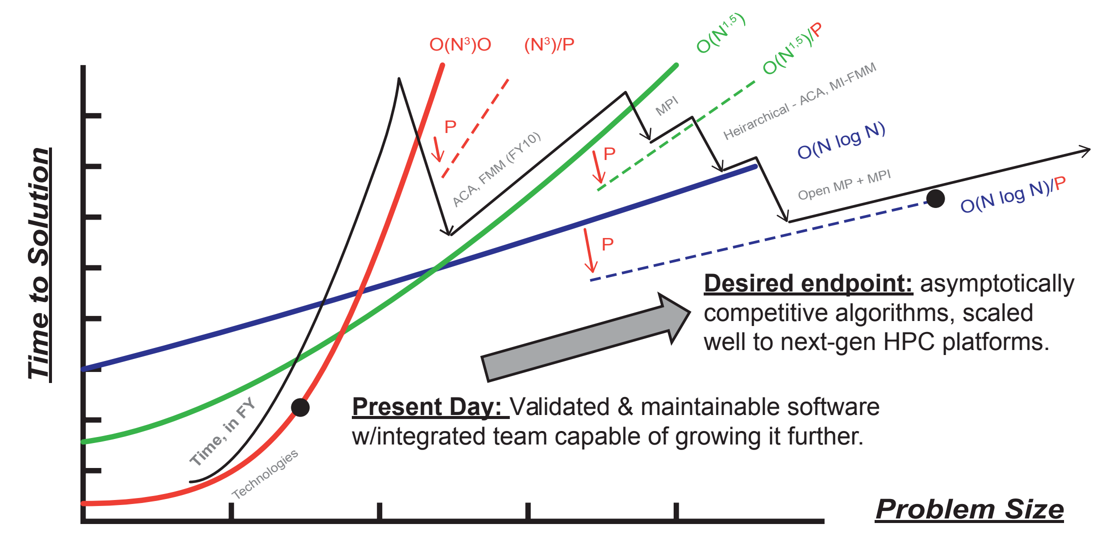

Figure 5. Schematic illustration of SENTRI scaling goals and techniques

desired level of modularity and quality. Immediately after beta release of SENTRI 1.0, a rewrite was begun to improve the modularity and software quality to provide the base for developing a highly scalable package that would be maintainable, extensible, and portable. The re-written code was released for beta testing in mid-FY2010 as SENTRI 1.5. The technical push in the next few years will be to improve the scaling from n^3 to $n^{3/2}$ to n log n by introducing a series of fast solver technologies (Figure 5).

The CREATE Air Vehicles Project released two products for beta testing in FY09, the fixed wing design tool KESTREL 1.0, and the rotorcraft design tool HELIOS 1.0 . The initial KESTREL release included the ability to model three different behaviors independently: 1) Fluid dynamics simulation for a single mesh, steady or unsteady, inviscid, viscous, laminar, or turbulent flow; 2) Coupled fluid dynamics and dynamics solution of a single mesh in rigid body motion, specified by either a user defined motion file or built up from user inputs describing sinusoidal or constant rate and hold pitch, yaw, or roll motions or combinations of these motions that include a pre-flight capability to check out mesh motion to ensure that motion is properly set up; and 3) Coupled fluid dynamics/structural mechanics simulation of a static position single mesh aeroelastic wing with second order temporal accuracy using a modal structural solver that includes a pre-flight capability to check out mesh quality during a forced elastic variation that may be encountered during a fluid-structure interaction simulation[9]. KESTREL 1.0 incorporates a rewritten version of AVUS (kAVUS) for the fluid dynamics. This package was able to compute solutions as accurately as typical fluid dynamics codes in common use (FLUENT, COBALT, FUN3D and USM3D). During beta testing, users with no prior experience with KESTREL demonstrated the ability to match flight data from F-16 and F-22 windup turns and other complex maneuvers (Figure 6).

- UC I => Ship Response To Standoff UNDEX Where Structure Remains Predominantly Elastic (minimal damage) (FSST)
- UC II => Ship Response to UNDEX & SURFEX Causing Moderate Structural Damage
- UC III => Ship Response To UNDEX & SURFEX Causing Severe Structural Damage (including SURFEX)
- UC IV => Ship Response To AIREX Causing Moderate Structural Damage
- UC V => Ship Response To AIREX Causing Severe Structural Damage
- UC VI => Ship Response To Unconventional Weapon Attacks

UC—Use Case: UNDEX—Underwater Explosion: FSST—Full Ship Shock Test: SURFEX—Surface Explosion: AIREX—Air Explosion

Table 2 USE CASEs for the 12 year NESM product development plan

The Helios 1.0 release provides the capability for four use cases, the ability to calculate: 1) Fuselage aerodynamics; 2) Fuselage with actuator disk model for the rotor; 3) Isolated

rotor in ideal hover and 4) Isolated rotor in forward flight with structural dynamics and trim. One major advance in Helios 1.0 is the ability to accurately calculate vortex shedding from the rotor tips through a combination of five body centered meshes for the three rotors, the fuselage and the rotor shaft, a fixed Cartesian mesh for the environment of the rotorcraft with a high order solver, and an accurate mesh interpolation method for connecting the body-centered meshes with the Cartesian background mesh[11]. This approach enables an accurate calculation of the vortex shedding through five or so rotations, sufficient to follow the vortex spiral from the rotors to below the fuselage. This ability to provide efficient, accurate vortex shedding predictions is a significant advance in the DoD's ability to calculate rotorcraft performance (Figure 7). It is beginning to enable engineers to analyze the fuselage loads due to the rotors and the associated vortices, and to optimize the performance of rotorcraft and improve their fatigue life. It is also useful for calculating the effects of vortex shedding from the front of aircraft on tail assemblies[12], the effects of vortices on acoustic noise production in aircraft, etc. The Helios team is working as part of the CREATE Program. However, it was started as one of the HCPMP institutes in 2006 and is still part of that program. The Helios team will formally join the CREATE program in 2012.

Several issues have arisen during the release process. One issue is user support. The CREATE tools are proving to be popular with the design community. There are over 100 volunteer beta testers. The need to support this large body of users while maintaining a focus on code development to deliver the 2010 releases has led us to develop an approach for code support that provides an adequate level of user support but also provides some filter of user requests and input to the developers. Without these filters the developers will spend all their time answering the telephone. We have borrowed some a number of support techniques developed by the commercial software industry. We have provided mechanisms for reporting bugs and issues to a web-site. An experienced support engineer examines the issue, and works with the user to develop a repeatable way to reproduce the bug or issue. The support engineer then passes that along to the developers. Simple problems are handled by the support engineer. In addition we are providing extensive user manuals and training, and a searchable blog or user forum for users to record their experiences and share them. Developers and users post solutions and comments to the forum.

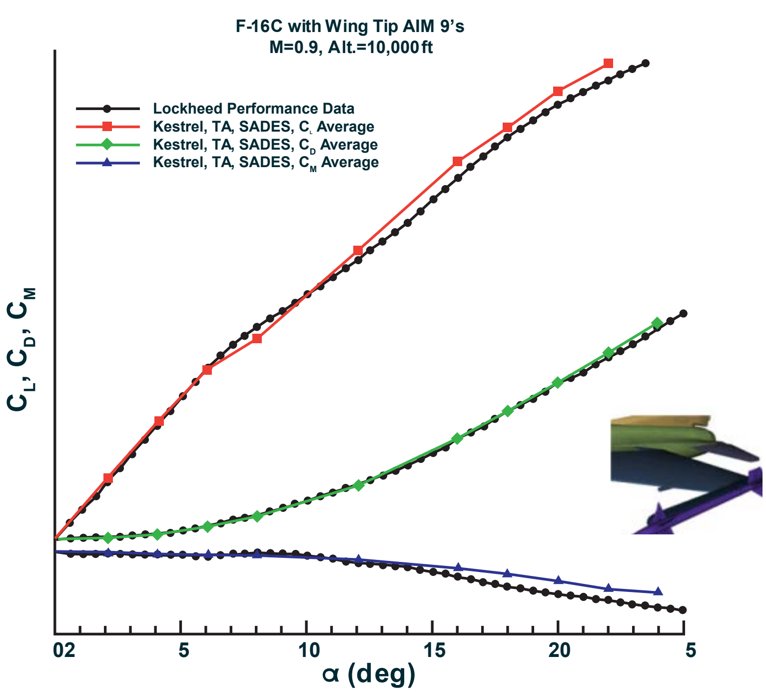

Figure 6. F-16C with wing tip AIM 9's at a Mach number 0.9 and an altitude of 10,000ft CL, CD, CM comparisons between Lockheed performance data and Kestrel CFD solutions[10]

Another issue is the need for the government to control distribution of the CREATE products and to ensure that the government also retains ownership of the software. The CREATE products are being developed to provide the US government a competitive advantage in military technology with respect to other countries. It is therefore imperative that the US retain the right to control distribution of the CREATE software. The US needs to own the CREATE tools to ensure that the tools are available to the government throughout the life-cycle of each product, that once the government pays for the development of the software, it retains ownership and doesn't have to pay additional funds for software that it paid to develop originally.

CREATE PLANS FOR 2011

Ten releases are planned for 2010-11 (Table 3). Five of these are enhancements of products released in FY09-10, and five are new products. For Air Vehicles, the new releases will be Firebolt 1.0 which will provide components to model propulsion systems for KESTREL and Helios. These will be delivered in FY2010-11 and incorporated into KESTREL 3.0 and Helios 3.0. DaVinci 1.0 will provide an initial set of low fidelity but fast physics-based design tools to develop conceptual designs of military aircraft from scratch[14]. While these won't have the physics fidelity of Kestrel and Helios, they will provide much better physics capability than existing design tools. DaVinci 2.0 and later releases will provide improved tools with better physics, higher accuracy, and automated design optimization.

Figure 7. Comparison of the existing state of the art for efficient vortex shedding calculations with legacy codes such as NSU3D and Helios.[13]

CREATE Ships project will release NavyFoam 1.0, a new, high-fidelity, scalable Reynolds-Averaged, Navier-Stokes hydrodynamics code for the calculation of resistance, drag, maneuvering, seakeeping and seaway loads for existing ships such as the DDG-1000, and for new and modified designs. The Ship project will also release the first version of its Rapid Design Integration Software, RDI 1.0. It will include algorithms for intelligent ship arrangements (ISA) (compartment layout), generation of optimized hull forms, and several multi-disciplinary design optimizers based on response surface models.

Finally the meshing and geometry project (MG) will deliver a set of modules (CAPSTONE 1.0) that will enable the other CREATE products to import, repair and cleanup existing geometries, and generate unstructured surface and volume meshes that are suitable for use in the analysis of air-vehicles, ships and RF antennas with platforms. The modules will include the ability to create geometric entities that will enable DaVinci to produce geometric models for conceptual-early design of air vehicles. Other modules will provide unstructured volume meshing technology that enables three dimensional analysis of aircraft, ships and RF systems. The project will deliver a new preprocessing platform that will provide a graphical front end for access to these modules.

CONCLUSIONS AND SUMMARY

The CREATE Program has begun to release useful software products 25 months after project start, and is set to release a full set of ten products each year starting in late 2010. CREATE has built ten successful product development teams, has developed initial requirements and validated them. It has developed initial plans for each product, and established a set of software engineering practices and processes, and a set of management processes. The program has also begun to engage the customer community and to provide support for the user community. It has also begun to address the issues associated with distribution and control of the CREATE products.

	Releases→FY09-10	Releases→FY10-11
AV	Kestrel 1.0	Kestrel 2.0
	Helios 1.0	Helios 2.0
		Firebolt 1.0
		DaVinci 1.0
Ships	NESM 0.1	NESM 1.0
	IHDE 1.0	IHDE 2.0
		NavyFoam 1.0
		RDI 1.0 (~6 components)
RF	Sentri 1.0, 1.5	Sentri 2.0
MG		Capstone 1.0

Table 3 CREATE Releases for FY09-10 and FY10-11

REFERENCES

[1] GAO, *DEFENSE ACQUISITIONS, Assessments of Selected Weapon Programs*, 2009, GAO-09-326SP,

[2] Council on Competitiveness, *High Performance Computing Initiative* (HPC), 2009,

[3] L. K. Miller, *Simulation-Based Engineering for Industrial Competitive Advantage*, 2010, Computing in Science and Engineering, **12**, 14-21

[4] S. Francis, *Warhead Politics, Livermore and the Competitive System of Nuclear Weapon Design*, 1995, UCRL-LR-124754, 247.

[5] H. Meuer, E. Strohmaier, J. Dongarra and H. Simon, *Top 500 Supercomputers*, 2010,

[6] K. Schwaber, *Agile Project Management with Scrum*, 2004, Microsoft Best Practices, 163.

[7] W.-M. Lin, M. Collette, D. Lavis, S. Jessup and J. Kuhn, *Recent Hydrodynamic Tool Development and Validation for Motions and Slam Loads on Ocean-Going High-Speed Vessels*, 2007, Proceedings of the 10th International Symposium on Practical Design of Ships and Other Floating Structures, Houston, Texas, September 30-October 5, 2007

[8] T. McGrath, *Computational Analysis of the Effect of Hydro-Reaction on Underwater Shaped Charge Warhead Performance Using the DYSMAS Hydracode*, 2009, Proceedings of the 80th Shock and Vibration Symposium, San Diego, CA, October 25 – 29, 2009

[9] S. Morton, D. McDaniel, D. Sears, B. Tillman and T. Tuckey, *Kestrel: A Fixed Wing Virtual Aircraft Product of the CREATE Program*, 2009, Proceedings of the 47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition, Orlando, Florida, Jan. 5-8, 2009, AIAA-2009-0338.

[10] S. Morton, D. McDaniel, D. Sears, B. Tillman and T. Tuckey, *Rigid, Maneuvering, and Aeroelastic Results for Kestrel - A CREATE Simulation Tool*, 2010, Proceedings of the 48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, Orlando, Florida, Jan. 4-7, 2010, AIAA-2010-1233.

[11] A. Katz, A. Wissink and A. Jameson, *A Multi-Solver Scheme for Viscous Flows Using Adaptive Cartesian Grids and Meshless Grid Communication*, 2009, Proceedings of the 47th AIAA Aerospace Sciences Meeting, Orlando FL, Jan. 2009, AIAA-2009-0768.

[12] N. Hariharan, J. Hunt, A. Wissink and V. Sankaran, *A Dual-Mesh Simulation Strategy for Improved AV-8B Aft-Fuselage Buffet Load Prediction*, 2010, Proceedings of the 48th AIAA Aerospace Sciences Meeting, Orlando, FL, January 2010 AIAA Paper 2010-1234

[13] V. Sankaran, J. Sitaraman, A. Wissink, A. Datta, B. Jayaraman, M. Potsdam, D. Mavriplis, Z. Yang, D. O'Brien, H. Saberi, R. Cheng, N. Hariharan and R. Strawn, *Application of the Helios Computational Platform to Rotorcraft Flowfields*, 2010, Proceedings of the 48th Aerospace Sciences Meeting, Orlando, FL, January 2010, AIAA Paper 2010-1230.

[14] G. Roth, J. Livingston, M. Blair and R. Kolonay, *CREATE_AV DaVinci: Computationally Based Engineering for Conceptual Design*, 2010, Proceedings of the 48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, Orlando, Florida, Jan. 4-7, 2010, AIAA-2010-1232.

DISTRIBUTION STATEMENT

Proceedings of the DoD HPCMP User Group Conference, June 15-17, 2010, Schaumburg, IL.

A. Approved for public release; distribution is unlimited 10-S-2649

DESIGNING AND DEVELOPING EFFECTIVE TRAINING GAMES FOR THE US NAVY

Talib S. Hussain, Bruce Roberts
BBN Technologies
Cambridge, MA
thussain@bbn.com, broberts@bbn.com

Ellen S. Menaker, Susan L. Coleman
Intelligent Decision Systems, Inc
Centreville, VA
menaker.ellen@idsi.com, coleman.susan@idsi.com

Kelly Pounds
IDEAS
Orlando, FL
kre8itnow@aol.com

Clint Bowers, Janis A. Cannon-Bowers
University of Central Florida
Orlando, FL
bowers@mail.ucf.edu, jancb@mail.ucf.edu

Curtiss Murphy, Alion Science and Technology
Norfolk, VA
cmmurphy@alionscience.com

Alan Koenig, Richard Wainess, John Lee
National Center for Research on Evaluation Standards, and Student Testing, UCLA
Los Angeles, CA
koenig@cse.ucla.edu, wainess@cse.ucla.edu, johnjn@ucla.edu

ABSTRACT

Game-based training offers great potential for providing low-cost training systems for learning cognitive and procedural skills within the U.S. Navy. We introduce an effort, sponsored by the Office of Naval Research, to harness, apply and harden this capability by creating validated training games for the Navy. Over the period of fourteen months, our multi-disciplinary team collaborated to develop and validate a flooding control training game to help students at the U.S. Navy Recruit Training Command (RTC) learn to be better sailors. The Flooding Control Trainer (FCT) provides individual training within the simulated interior of an Arleigh-Burke class destroyer. The trainer reinforces damage control skills that the recruits have been exposed to in lectures, but which they have not had a chance to practice in context. In developing the trainer, we focused on both the specific application domain as well as the design methods required to ensure that the trainer was based on relevant learning objectives, incorporated a strong narrative, used an instructional strategy and a game play style that were complementary, and contained embedded assessment capabilities. The FCT is based on the open-source Delta3D engine. To support effective training, we augmented the engine with a task-based instructional infrastructure and a variety of feedback mechanisms, including real-time guidance and feedback as well as after-action debrief. We conducted several empirical tests of the product, including a usability study and a learning validation study using the target recruit population as subjects. The results indicate that the FCT is usable, well-received by recruits and produces a significant improvement in performance across a range of cognitive and procedural skills, including situational awareness, communications, navigation and decision-making. We present our approach, describe the training game design, discuss the studies conducted and their results, and discuss next steps to create Navy training games for use beyond RTC.

ACKNOWLEDGEMENTS

The research reported in this paper was conducted under the management of Dr. Ray Perez under the Office of Naval Research contract number N00014-08¬C-0030. We would like to thank our customer Dr. Rodney Chapman, Chief Learning Officer of the Naval Service Training Command, and his staff for their active participation in refining and guiding our product design, and Lt. Greg Page from the Recruit Training Command for his assistance in providing subject matter expertise. We would like to thank Dr. Ray Perez and Mr. Paul Chatelier for identifying opportunities for putting our research efforts in context of the Navy's current needs. We would like to thank Steve Greenlaw of Intelligent Decision Systems, Inc. for providing subject matter expertise. We would like to thank the software designers and engineers who contributed to the development and testing of the product: Brad Anderegg, Chris Rodgers and

David Guthrie from Alion Science and Technology; Rachel Joyce, Lucas Blair and Julian Orrego of UCF; Kerry Moffitt, Todd Wright, John Ostwald and Wallace Feurzeig of BBN Technologies. Finally, we would like to thank our former teammates Erin Panttaja, Vance Souders and Jason Seip, as well as our new teammates AhnNa Yi and John Paulus of CHI Systems, Inc. who are contributing to the ongoing enhancement of our training system.

ABOUT THE AUTHORS

DR. TALIB S. HUSSAIN is Senior Scientist at BBN with a broad interest in learning and training for both machines and humans. He is currently co-Principal Investigator on the ONR-sponsored Tools for Games-Based Training and Assessment of Human Performance project, which is investigating advances in authoring technology for pedagogically strong game-based training systems and is developing game-based training solutions to support Navy recruit training. He was recently the development lead on the DARPA¬ sponsored Plan Order Induction by Reasoning from One Trial (POIROT) project, which is applying a broad range of machine learning and artificial intelligence techniques to learn procedural knowledge based on a single observation of a human performing a task. He was the principal investigator on the DARPA-sponsored Gorman's Gambit project, part of the DARWARS program, which studied the design issues involved in training military teamwork skills using modern multi-player game (MPG) technology. Forty infantrymen from Fort Benning, GA participated in a "capture-the-flag" exercise developed as a scenario within a COTS MPG. Valuable lessons were learned on the requirements for measuring team performance within a game.

BRUCE ROBERTS is a Lead Scientist at BBN Technologies, where he has developed numerous simulation-based intelligent tutoring systems: for Air Force technicians troubleshooting flight-line avionics, for conning officers practicing shiphandling in a virtual environment, and for Air Weapons Officers controlling air-to-air attack aircraft. He is currently co-Principal Investigator on the ONR-sponsored Tools for Games-Based Training and Assessment of Human Performance project, which is investigating advances in authoring technology for pedagogically strong game-based training systems and is developing game-based training solutions to support Navy recruit training. Recently, he was the Principal Investigator for DARWARS architecture and integration, part of DARPA's Training Superiority program, and led the rapid development and successful deployment of DARWARS Ambush!, a widely used multi-player game-based training system.

DR. CLINT BOWERS currently resides on the University of Central Florida's Psychology Faculty as well as being Co-Director of the RETRO (Research and Emerging Technology Research Organization) Lab at UCF's Institute for Simulation and Training. Clint's current research interest is in the use of technology to facilitate teamwork. This takes several forms. The first includes basic research on the nature of effective teamwork and the factors that influence it. A second research thrust involves the training of teams. Clint is especially interested in the use of training technologies and simulation in training team skills. Finally, Clint is interested in the use of technology to assist teams in their day-to-day tasks. This includes research in information visualization, groupware, and other hardware/software systems. Besides his interest in team performance, Clint is also involved in several student projects. These include the use of virtual reality to teach deaf children, the neuropsychology of spatial abilities, and the use of warnings to manage predictable non-compliance.

DR. JANIS A. CANNON-BOWERS holds M.A. and Ph.D. degrees in Industrial/ Organizational Psychology from the University of South Florida, Tampa, FL. She recently left her position as the Navy's Senior Scientist for Training Systems to join the School of Film and Digital Media at the University of Central Florida, Orlando, FL as an Associate Professor. As the team leader for Advanced Training Research for the Navy, she was involved in a number of research projects directed toward improving performance in complex environments. These included investigation of training needs and design for multi-operator training systems, tactical decision-making under stress, the impact of technology on learning and performance, training for knowledge-rich environments, human-centered design, human performance modeling and development of knowledge structures underlying higher order skills. At UCF, Dr. Cannon-Bowers is continuing her work in technology-enabled learning and human performance modeling. Her goal is to leverage and transition DoD's sizable investment in modeling, simulation and training to other areas such as entertainment, workforce development and life-long learning and education. To date, she has been awarded several grants to support this work, including a recent award by the National Science Foundation under their Science of Learning Center program. Dr. Cannon-Bowers has been an active researcher, with over 100 publications in scholarly journals, books and technical

reports, and numerous professional presentations. She is a Fellow of the Society of Industrial and Organizational Psychology (Division 14 of the American Psychological Association).

DR. SUSAN L. COLEMAN, CPT, is the Chief Performance Officer for Intelligent Decision Systems, Inc. Dr. Coleman oversees all phases of human performance analyses, while specializing in instructional systems design processes. She earned a Ph.D. in Instructional Technology and Design and has analyzed performance and designed and developed performance solutions since 1983. She spent the last 18 years analyzing and designing training systems for the military. Dr. Coleman conducts instructional design research, training effectiveness evaluations, design analyses, technology integration front-end analyses, and human performance improvement analyses.

DR. ALAN KOENIG is a Senior Research Associate at UCLA/CRESST where he specializes in the application of innovative uses of technology for delivering and assessing instruction. His research focuses on the design and implementation of computer-based games and simulations designed for classroom and/or military training environments. His current work centers on the development of automated assessment systems for high fidelity games and simulations used in the military. Prior to joining CRESST, Dr. Koenig spent 10 years working in the technology sector as both a software developer and mechanical design engineer. Dr. Koenig holds a PhD in Educational Technology from Arizona State University, a BS in Mechanical Engineering from the University of Hartford, and a BA in Economics from the University of Connecticut.

DR. JOHN LEE'S current research is related to technology-based assessments in a variety of Navy/Marine Corps contexts. He is currently working on the development of a computer-based assessment tool for assessment of Tactical Action Officers (TAO) in a simulated CIC (Combat Information Center) onboard Navy ships called the Multi-Mission Team Trainer (MMTT). He is also working on a simulation-based re-certification assessment of marksmanship coaches' fault checking abillity that delivers just-in-time, individualized instruction that utilizes Bayesian nets for diagnosis and remediation. A third project he is also working on is a game based assessment project for the Navy related to assessment of complex skills (starting with damage control) also using Bayesian nets for real time and after action assessment of skills including situation awareness, decision making and communication. His research interests include data-informed decision making, knowledge mapping, simulation-based assessments, and distance learning.

DR. ELLEN S. MENAKER, CPT, is the Chief of Research and Evaluation for Intelligent Decision Systems, Inc. Dr. Menaker has over 30 years of experience in the fields of research and evaluation, cognitive development, and human performance. Dr. Menaker oversees the design, data collection, and analysis phases of research; and various types of analyses and evaluations. She specializes in learning theory, measurement, and instructional systems design. Her academic and industry experiences include conducting research for various military, governmental, and educational entities. Recent studies have focused on implementation of learning strategies, including use of massive multiplayer online games (MMOGs) and the integration of SCORM modules into gaming environments. Recent literature reviews have focused on the application of learning theory, strategies for developing situational awareness, and identification of foundational skills for lifelong learning. Dr. Menaker is an active member of AERA, ISPI, and she serves on the Education Committee for IITSEC.

CURTISS MURPHY is a Project Engineer in the AMSTO Operation of Alion Science and Technology. He manages the game-based training and 3D visualization development efforts for a variety of Marine, Navy, and Joint DoD customers. He is a frequent speaker and author and specializes in open source technologies such as the Delta3D Game Engine (www.delta3d.org). He has been developing and managing software projects for 17 years and currently works in Norfolk, VA. Curtiss holds a BS in Computer Science from Virginia Polytechnic University.

KELLY POUNDS' multi-dimensional career began as a schoolteacher in Orange County (FL) Public Schools where she spent twelve years as a classroom teacher. Kelly next took her passion for learning and instructional design into the corporate environment as a Technology Designer for the Disney University where she developed computer-based and classroom training products for leaders at the Walt Disney World® Resort. Her next corporate role was at Hard Rock Cafe International as Manager of Organization Development where, as an innovative strategist, she helped the corporate leadership collaboratively develop strategic planning processes and products. Kelly's enthusiasm for learning is currently shared through her role as

Vice President of i.d.e.a.s. Learning where she puts her past work experience and her master's degree in Educational Technology to work everyday. i.d.e.a.s. is an innovation studio employing its core competency of storytelling to create entertainment, marketing, and learning products.

DR. RICHARD WAINESS is a senior researcher with the National Center for Research on Evaluation, Standards, and Student Testing (CRESST) in the University of California Los Angeles' Graduate School of Education and an adjunct associate professor in the University of Southern California's Rossier School of Education. Dr. Wainess' research interests center on the use of games and simulations for training and assessment of adult learners. His most recent work is focused primarily on assessment of problem solving and decision making using computer-based interactive tools. He has authored and co-authored numerous reports, articles, and book chapters and has presented at many conferences on the topic of games and simulations for learning, with a particular emphasis on instructional methods, cognitive load theory, motivation, and learning outcomes. Richard has a B.A. in Radio-Television Production, an MS.Ed in Instructional Technology, and a Ph.D. in Educational Psychology and Technology.

INTRODUCTION

Immersive training environments based on modern computer game engines are increasingly being used within the U.S. and other militaries to provide training on a range of skills (O'Neil & Perez, 2008; Roman & Brown, 2008), including team convoy operations (Diller et al., 2004; Roberts et al, 2006), route clearance (O'Bea & Beacham, 2008), operationally relevant language skills (Johnson et al., 2007), and small unit tactical operations and mission rehearsal (McDonough, J.P. & Dmochowski, N., 2008). To date, most game-based military training systems have focused on ground or air based operations. However, game-based technologies have the potential to provide valuable training on the ship-based operations of the U.S. Navy. The Office of Naval Research is currently funding research to identify the learning benefits of game-based trainers for naval training, harden the technology and methods for developing game-based training for the Navy, and produce effective training systems that can be deployed in the near term.

We introduce a training game developed to reinforce the skills needed in controlling flooding on board a ship. The Flooding Control Trainer (FCT) was developed in collaboration with the Naval Service Training Command to serve three purposes: (1) supplement classroom instruction at the Navy Recruit Training Command (RTC) to produce better-trained recruits; (2) reduce the demand on current anticipated training resources, and (3) produce a training platform that can be used across the Navy technical schools and, eventually, in the fleet.

Over the period of fourteen months, starting in February 2008, our multi-disciplinary research and development team conceived, developed and validated a 3D immersive training game for flooding control. The FCT is built using the open-source Delta3D game engine (**www.delta3d.org**, Darken et al., 2005, Murphy et al. 2008) and runs on a typical modern desktop computer. During our effort, a strong emphasis was placed upon drawing together the best practices of training game design from several disciplines, including instructional system design, narrative design, formative and summative assessment, and game design and development. In this process, we identified a number of lessons learned and made good progress towards identifying cohesive and consistent design and development methods for game-based training systems (Hussain et al., in press).

The FCT provides individual training on flooding control skills within the interior of a simulated Arleigh-Burke class destroyer. It focuses on learning objectives that directly support the RTC curriculum and embeds the training within a story that promotes core values, provides a relevant context and reinforces the culture of the service. The game reinforces decision-making, communication protocol, flooding control procedures and situational awareness. It provides information resources and feedback designed to help students of different levels and backgrounds succeed.

To ensure the relevance and success of the training game, we conducted multiple tests using the target recruit population as subjects, and enhanced the system iteratively based on the results. In April 2009, we conducted a validation study that demonstrated learning transfer from the game environment to a real physical environment. Significant performance improvements were observed across almost all the skills taught in the game.

We present some details on the design approach used, introduce the final game design, discuss the studies conducted and their results, and discuss our next steps.

BACKGROUND

Navy Recruit Training

The U.S. Navy's boot camp, the Recruit Training Command (RTC) at Great Lakes, Illinois, currently trains 40,000 recruits per year, drawn from diverse socioeconomic, cultural and educational backgrounds. Recruits undergo eight weeks of training, delivered primarily via classroom lectures and drill instruction by recruit division commanders. There are some computer based training labs and a few hands-on training labs, such as fire fighting, using self-contained breathing apparatus and line handling. At the end of their training, recruit performance is evaluated in an intense ten-hour capstone evaluation event, Battle Stations 21 (BS21). BS21 is a physical simulation, in a building, of an Arleigh-Burke class destroyer, complete with a simulated dock and ship exterior, as well as several internal decks. The ship is named the U.S.S. Trayer. During the event, recruits complete seventeen different training scenarios, ranging from routine (moving stores, standing bridge watch) to critical (flooding control, fire fighting) to extreme (dealing with injured shipmates during a mass casualty caused by an explosion).

The scale and diversity of the RTC training population provide ongoing training challenges. Currently, recruits face significant cognitive overload in BS21 due to limited opportunities to practice the skills they have been taught, spatial disorientation from never before having been on board a ship, the need to learn significant new material once they are in the BS21 exercise, and a high degree of stress due to being evaluated on unfamiliar skills (HPC, 2008). In particular, these issues can be seen in damage control situations. Further, the recruit population's diversity is increasing with time and current requirements call for an increase to over 46,000 recruits per year by 2011 with no increase in funding. As a result of these challenges, RTC is exploring the use of advanced training technologies, such as game-based training, to augment and enhance the training they provide.

Battle Stations 21 represents a high performance environment, characterized by rapidly evolving and changing scenarios, severe time pressure, serious consequences for error, command and peer pressure, fatigue, and a need for complex coordination of action. As such, BS21 requires highly complex performance that combines both individual and team level skills.

Research into optimizing performance in such environments has been ongoing for several years. Overall, findings suggest that, in order to be successful, individuals must be able to execute mission essential skills quickly and without hesitation. More specifically, the research literature suggests:

1. Complex performance must be broken down into requisite components so that individual knowledge and skills can be isolated and trained to proficiency before introducing the full complexity of the task. (Goldstein, 1993)

2. Under stress, performance is most resilient when it becomes automatic or habitual. This can be accomplished most efficiently by allowing trainees to practice until skills are over-learned (i.e., practiced beyond the point where performance is learned so that it becomes habitual and requires little active cognitive processing for successful accomplishment). (Kirlik et al., 1998)

3. Training for complex skills requires hands-on practice and feedback to be most effective. (Salas & Cannon-Bowers, 2001)

4. Synthetic learning environments, including simulations and games, are excellent environments in which to provide learners with realistic tasks so that they can practice essential skills. (Cannon-Bowers et al., 2008)

5. Trainees who are confident in their knowledge and skills are more likely to perform without hesitation. (Wurtele, 1986)

Game-Based Training Technology

The use of games and game-based technology for education and training has been increasing over the past few years (O'Neil & Perez, 2008; Smith, 2008). Computer game-based training systems share a number of potential characteristics with effective instructional tools and, therefore, have a great potential to affect learning. For example,

1. Games provide interactive experiences in a task-based environment with repeated exposure to important cue patterns. This is consistent with the development of expertise (Glaser, 1989; Chi et al., 1988; Bransford et al., 1999), anchored instruction (e.g., Bransford et al., 1990; CGTV, 1992; 1997; 2000) and active learning (Rothman, 1999; Chi, 2000; Mayer, 2001; Vogel et al., 2006).

2. Games provide a model-based world in which students may manipulate variables, view phenomena from multiple perspectives, observe system behavior over time, draw and test hypotheses and compare their mental models with representations in the external word. These features are consistent with the model-based reasoning concepts advocated by learning researchers (Gentner, 1983; Raghavan et al., 1997; 1998; Leher & Schauble , 2000; Cartier & Stewart, 2000; Zimmerman et al., 2003; Stewart et al., 2005).

3. Games provide successive tasks to help players make progress towards concrete, specific and timely goals. Goal setting in instruction enhances learning (Locke et al., 1981; Locke & Latham, 1990; Schunk & Ertmer, 1999).

4. Well designed training games also provide a variety of elements that can enhance motivation and learning, such as a sense of accomplishment (Bandura, 1977; 1986; Gist et al., 1989; 1991); informative feedback (Bransford et al., 1999; Salas & Cannon-Bowers, 2000) and a sense of challenge or competition (Epstein & Harackiewicz, 1992; Reeve & Deci, 1996).

Hence, properly designed training games can provide engaging learning environments that result in high time-on-task, reproducible learning outcomes and low human and system resource requirements.

However, the empirical evidence supporting the effectiveness of games for learning has generally been mixed (O'Neil et al., 2005). This has been due largely to a poor understanding in the field of how to effectively design games to support training (Gunter et al., 2006; Hussain & Ferguson, 2005; Hussain & Feurzeig, 2008), and an associated lack of empirical evidence for what effects different elements of a game-based training system have upon learning outcomes (Wilson et al., 2009). Some evidence has been presented indicating that game-based technology "is most effective as part of a blended training solution," and that using games for mission rehearsal prior to undergoing live training "makes live training more effective and efficient" (Roman & Brown, 2008). However, very few studies have been conducted to demonstrate the reliable transfer of learning, in a military domain, across a range of cognitive and procedural behaviors from a game-based training system to a physical environment.

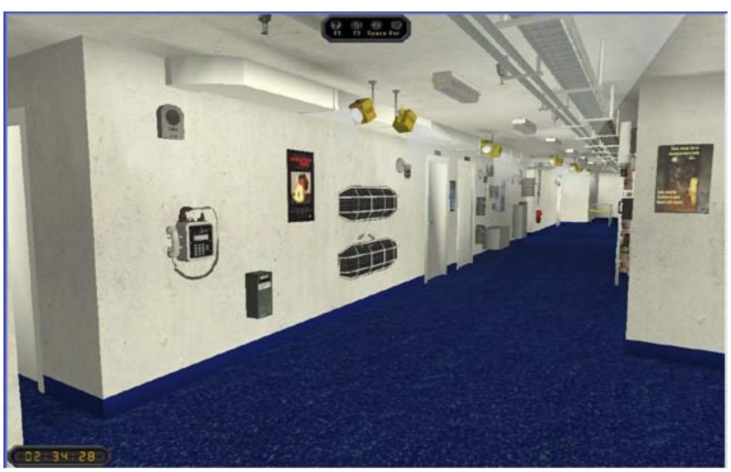

Figure 1: Simulated ship interior

Using modern learning theory as a basis, we put forth that game-based training should be able to create a strong positive learning effect with minimal instructor involvement. To prove this hypothesis, we developed the Flooding Control Trainer using proven instructional principles and modern game design. We then ran U.S. Navy recruits through a near transfer study to validate our hypothesis. The results were both compelling and conclusive.

FLOODING CONTROL TRAINER OVERVIEW

In order to address the training challenges in the Navy and at RTC in particular, we developed a prototype training game that teaches the basic skills necessary to control flooding on board a ship. Over the course of fourteen months, starting in February 2008, we designed, developed and validated the Flooding Control Trainer. Details on the design and development process, including lessons learned, are given in Hussain et al. (in press).

Gaming Experience

The FCT is a single-player game that uses a first-person perspective. The 3D virtual environment models the interior of an Arleigh-Burke class destroyer with compartments of different types and appropriate fixtures and equipment (see Figure 1). The passageways resemble those of the U.S.S. Trayer at BS21. The player navigates the ship in a first-person perspective using the mouse and keyboard. The player interacts with the virtual world using the mouse to perform typical game actions, such as opening doors, inspecting objects, collecting personal protective equipment, and using damage control tools. There are no animated characters in the game, but

the player can interact with virtual characters via dialogs over a communications device. Dialogs appear as a pop-up window in which the player can respond to a virtual character by choosing from several alternatives (see Figure 2).

The game-play is based on completing a sequence of missions. There are three missions in the game, including a tutorial that teaches the player how to play the game itself. Each mission starts with a briefing that gives the player the key mission goals and puts them in the context of the underlying narrative. Each mission requires the player to achieve a set of tasks related to damage control. Some of these tasks are given to the player at the beginning of the mission and some are given during the course of the mission based on their actions. During a mission, the player has relatively free rein to interact with the virtual world. However certain actions may be prevented until the player has completed a particular task. Progress in the game is achieved by completing tasks that have been assigned to them.

Figure 2: Dialog mechanism, providing multiple response options from which to select

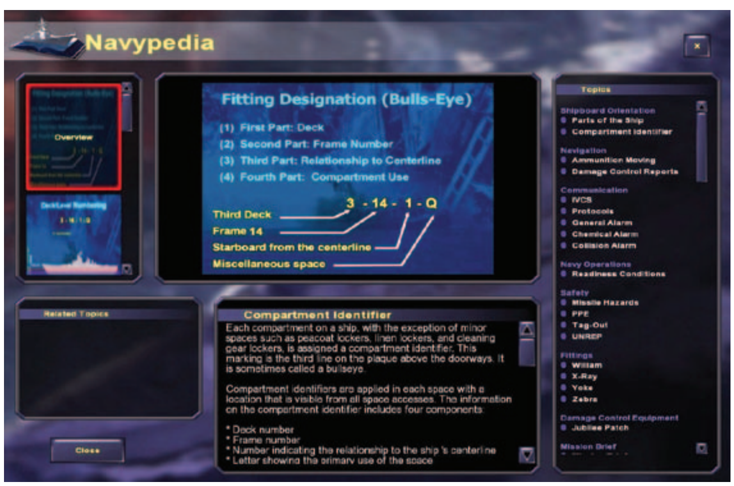

Figure 3: Navypedia mechanism, providing didactic content within the game on request

Learning Experience

The FCT is designed to be played without assistance from a human instructor. It weaves instruction throughout the gaming experience, and varies its instruction based on the performance of the student. Different missions focus upon different learning objectives, and successive missions get increasingly complex and difficult. In each mission briefing, the student is explicitly given directions that relate to the learning objectives for that mission. During a mission, the player is given guidance and feedback based on their actions. Content support is provided to players through access to the "Navypedia" help system (see Figure 3). For example, they can go to the Navypedia to find out what safety equipment they need when fighting a flood. Critical errors can result in penalties or failure. At the end of each mission, a debriefing on their performance is given.

Technology

The FCT is built using Delta3D, an open-source game engine, and an open-source instructional logic engine also developed as part of the project. The instructional logic engine is part of a platform-independent pedagogy middleware that interacts in real-time with the game engine to control the instruction. It communicates game events to a separate process which executes explicitly defined instructional and assessment logic and, in turn, directs certain pedagogical actions within the game, such as providing feedback. While details of the middleware are beyond the scope of this paper, a key capability to note is that the instructional logic is authorable using a visual logic editor that supports rapid prototyping and modification of the instruction.

TRAINING DOMAIN

The key training goals of the system are to:

- Develop flooding control skills
- Develop cognitive skills and a sound and robust mental model in the areas of situational awareness, communication and decision-making
- Establish patterns for adaptive thinking

Further, a key ability that applies across all shipboard activities is an understanding of one's location on the ship and navigating efficiently between different locations. Due to the spatial disorientation experienced by students in BS21, training shipboard navigation was an additional training objective. The FCT targets students that are novices who have declarative knowledge about ships and basic damage control procedures. The FCT design assumes that students have been through formal training to acquire the declarative knowledge, but that they have not been required to apply this information or to draw on this information to solve problems. In other words, the recruits have classroom instruction, but very little hands-on experience with flooding control and related skills. The expectation is that learners may not fully understand the information and that they require practice in realistic contexts to build sound mental models (HPC, 2008).

As such, the FCT focuses upon novice-level skills and upon reinforcing the types of decisions a novice would make in the Navy fleet. On board a ship, key damage control decisions are made by personnel in Damage Control Central (DCC). DCC receives damage reports from across the ship and has the key responsibility to coordinate repair operations and ensure the safety of the ship as a whole and its sailors. In general, DCC will make key decisions regarding how to combat a particular casualty, based on a thorough understanding of the affected systems and a higher-level understanding of what is happening on the ship. In particular, certain key actions require permission from DCC due to the potentially negative consequences to the sailor and/or to the ship. It is a role requiring significant expertise, and hence decisions specific to DCC are not trained in the game. Rather, emphasis is placed upon the student's interaction with the environment, use of repair equipment, and communication with superiors and DCC.

The flooding control skills required of a sailor include the actions and processes followed in preparation for a potentially dangerous situation, the ability to detect and identify flooding situations, the communication and coordination skills required to ensure that appropriate members of the ship are appropriately informed at appropriate times, following correct personal and ship safety protocols, and following the correct repair and follow-up procedures.

In a flooding situation, it is important to identify the source and type of the flooding. For example, the flooding may be due to a leaking pipe or a hull breach; the fluid may be fresh water, salt water, oil or fuel; and the flooding may range from minor to severe. Different types of damage require different types of repairs, such as patching a pipe, plugging a hole or shoring up the hull. It is also critical to maintain the watertight integrity of the ship. Throughout the ship are watertight doors. In a flooding situation, it is important to keep those doors closed to avoid the risk of a flood spreading through the ship. As a result, opening certain doors may require permission from DCC.

In addition to these basic flooding control skills, it is also important for a sailor to understand general damage control processes, general communication skills and protocols, and the relationship between their actions and events elsewhere on the ship. To ensure their safety in a real or potential damage control situation, sailors must always don the appropriate personal protective equipment (PPE). PPE can include boots, gloves, helmets, fire fighting gear, breathing gear and more depending on the situation. For communications in particular, clarity and accuracy is critical. To avoid miscommunications, all interactions should include the identities of the participants, and all instructions should be confirmed by repeating them back verbatim. In addition to mitigating interference caused by noise and other activities, repeating back instructions ensures that no misunderstanding has occurred and provides an opportunity for correction.

On a Navy ship, a specific coordinate system is used to identify locations. All doors, compartments and passageways, as well as certain equipment, are identified using three coordinates that indicate their deck, frame number (position forward to aft) and position port to starboard.

Learning Objectives

The game reinforces a basic "Assess-Report-Act" approach to flooding control. Three high-level categories of learning objectives related to cognitive readiness are used as an organizing principle.

- Situational Awareness: The ability to recognize cues, interpret cues and predict consequences.
- Communication: The ability to know whom to contact, when to contact, and how to report.
- Decision-making: The ability to follow appropriate protocols, follow orders and take initiative to complete a mission.

Table 1 summarizes the specific terminal and enabling objectives of one of the missions in the Flooding Control Trainer, organized by cognitive readiness categories.

TRAINING SYSTEM DESIGN

Design Overview

A highly iterative design process was followed (Hussain et al., in press) in developing the Flooding Control Trainer, during which a variety of different instructional, narrative, assessment and gaming elements were considered. The final training system has four key components. The first component is a supporting story that evolves as play progresses. The second component is an environment with high physical fidelity (realistic simulation of DDG ship interior) and simple interaction methods that were tested for usability. The third component is a progression of increasingly challenging game levels, each forming a distinct "mission" to be accomplished by the student. Specifically, the FCT contains three missions, one of which provides a tutorial on how to use the game and two others which provide training on navigation and flooding control skills. The fourth component is a rich suite of mechanisms and associated instructional logic that provide different types of guidance and feedback to the student under different situations and performance outcomes.

Narrative

The game's backstory begins following graduation from RTC as the sailor is posted to an Arleigh-Burke class destroyer. The ship's mission is to provide support in the Middle East. The ship has left port and is approaching its station near Abu Dhabi. Upon starting the game, an introductory movie (i.e., cut-scene) relates this backstory. At the end of the cut-scene, the student is encouraged to behave with Honor, Courage and Commitment, the core values of the U.S. Navy.

Terminal Objective	Enabling Objective
Situational Awareness	
Recognize abnormal condition	Use cues to detect flooding situation
Assess flooding situation	Recognize source and type of leak
Recognize shipboard navigation cues	Recognize and interpret compartment identifiers
Anticipate consequences of actions	Anticipate consequences of securing the fire main valve
Communication	
Recognize when situation warrants communication	Report flooding conditions in a timely manner
	Request permission to enter compartment
	Report when repair actions are complete
Report appropriate information	Report relevant information to DCC
Report information accurately	Report information accurately to DCC
Repeat back accurately	Repeat back DCC instructions accurately
Decision-Making	
Maintain watertight integrity	Enter compartment with permission
	Secure compartment doors as required
Follow safety protocols	Don appropriate PPE
Take proper actions to combat flooding	Follow directed orders
	Select correct patch repair items
	Position patch properly during application
	Use wrench to tighten patch
Use compartment identifiers to navigate ship	Successfully navigate using compartment identifiers

Table 1: Learning Objectives

	Purpose
Embedded information	Enable the student to examine objects in the environment and receive information about those objects and how to use them (mouse-over, inspect).
Advance priming	Provide students with directions and a summary of learning goals prior to start of mission ("briefing")
Current objectives list	Provide students with explicit information about the mission objectives they should be pursuing and which objectives they have achieved.
In-dialog hints	Use the natural dialog of the game to provide the student with suggestions or detailed information
Explicit instructions	Use a pop-up suggestion to explain the details of a procedure to the student prior to performing a task involving that procedure.
Explicit cues	Use a pop-up suggestion provide short cues or questions to promote thinking on part of student
In-game priming	Provide students with a just-in-time reminder to ensure they enter the event focused on the correct behaviors.
Visual aid	Provide students with a minimap to assist in spatial orientation
Didactic reference	Provide students with access to written and visual explanations of different aspects of ship operations and damage control ("Navypedia")

Table 2: Instructional Guidance Mechanisms

Feedback mechanism	Purpose
On screen cumulative performance bars	Immediate implicit performance feedback. A green merit bar increases as tasks are completed. A red demerit bar increases as errors are made. When the demerit bar reaches maximum, a failure occurs.
Natural consequences	Demonstrate the consequences of an error without ending gameplay and implicitly show why performing correctly was important.
Non-interrupting feedback	Alert student to a positive or negative behavior using the natural interactions of the game (e.g., dialog with DCC).
Interrupting feedback	Alert student to performance above expectations or critical errors and interrupt gameplay to ensure that students receive specific information explaining the alert.
Catastrophic end of the level	Teach the student that the behavior that caused the catastrophic event is not acceptable in any way. This is reserved for actions that can bring immediate/fatal danger to self or ship.
Ranking	Provide student with a rank (out of 5) indicating performance against ideal.
Debrief at the end of level	Explicitly summarize strengths and weakness of the student's performance and provide appropriate guidance.

Table 3: Instructional Feedback Mechanisms

The training is delivered in multiple game levels. To reinforce military protocol, each level is given as a mission to perform. Each mission begins with a briefing relating the objectives of the mission, and ends with a debriefing summarizing what happened in the mission.

The first mission (the tutorial) begins in a heightened state of readiness as the ship is preparing for an underway replenishment (UNREP). During the first and second missions, the student helps prepare the ship for UNREP by securing compartments, verifying the status of equipment and moving equipment. During the underway replenishment, however, a collision occurs between the two ships. A second cut-scene is used to show the collision as it happens, with the goal of making the story tensions more apparent and immersive. As the third mission starts, the ship is at general quarters and the sailor is ordered to investigate potential flooding. As the mission develops, a leaking pipe is discovered that requires patching.

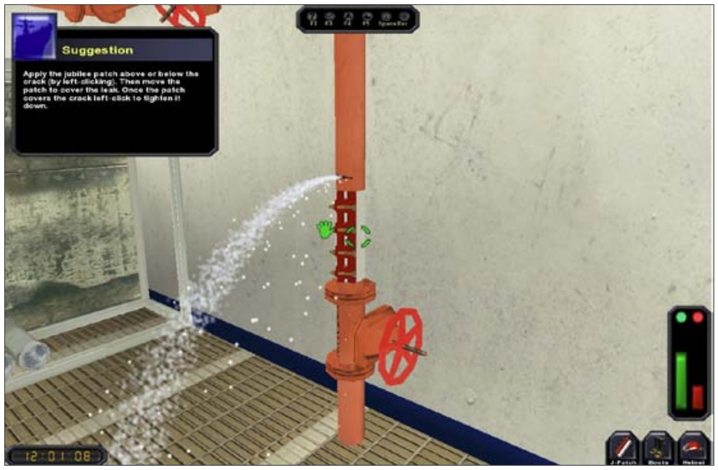

Figure 4: Example of a suggestion being provided in a pop-up window (see top left)

Figure 5: Example of a demerit message being given in a pop-up window (see bottom-right)

This narrative arc was chosen to make the training experience highly relevant to our target audience, to provide motivation by stressing the core principles of the U.S. Navy, and to provide a context in which a wide variety of casualties could occur. In addition to using cut-scenes to introduce and develop the story, dialogs within a mission are also used to advance the story.

Instructional Design

The Flooding Control Trainer, generally, applies a guided discovery instructional strategy. In such a strategy, it is important to balance the desire to give students explicit advance information to ensure they are properly prepared for the events they encounter with the desire to allow the students to learn on their own by making mistakes and using feedback after the fact to ensure they reflect appropriately on their performance. It is also important, as a game-based trainer, to use instructional interactions with the student that seek to avoid interrupting the flow of the game or providing information that is out of context with the narrative and immersive context of the game.

The FCT uses a variety of methods to communicate instructive information to the student. In general, our approach is to provide both non-interruptive and interruptive guidance and feedback of varying detail depending on the nature of the mistake made while keeping the player immersed in the story as much as possible (see Tables 2 and 3). Some situations requiring new skills may result in some unsolicited guidance provided in a small pop-up window (see Figure 4). Minor errors will generally result in some feedback in the same window. The guidance and feedback can take the form of a hint, question or instruction. Conversely, when the student makes a critical error, a demerit will interrupt gameplay visually and aurally with a specific message about the error and a warning sound (see Figure 5). For both hints and demerits, the initial message will be somewhat general. If an error is repeated, subsequent messages will be more detailed. Errors of omission or delay, as well as errors of commission are addressed.

Throughout a mission, the student's actions are assessed to determine whether they are demonstrating appropriate intent and/or accuracy. The student's performance against every terminal objective is assessed automatically via the student's actions in the game and choices in dialogs. Dialog interactions form a key method for assessing user performance against a variety of communication and situation awareness learning objectives. These assessments are context-sensitive (i.e., the same dialog choice may be correct or incorrect depending upon prior user actions). A single dialog interaction can result in errors against different objectives (e.g., reporting appropriate versus accurate information), and different types of feedback (e.g., dialog responses versus demerits).

Figure 6: Example of a cut-scene showing the catastrophic consequences of a critical error

The FCT uses scaffolding techniques to minimize cognitive load while providing effective practice and training on the learning objectives. In the earlier levels, the student is provided with a fairly limited number of gameplay options and is constrained to follow a highly linear path through the mission tasks. In the later levels, the students are allowed increased free-play and some mission tasks may be varied in order. For example, in the tutorial, the passageways are blocked to prevent the student from going too far from the initial location and getting lost. In the third mission, the student has free rein of the ship.

Finally, the FCT gradually introduces complexity in order to minimize cognitive load. In the earlier levels, the student has few tasks to perform and cannot make any critical errors leading to failure (though they can make too many minor errors and thereby fail). In the later levels, the student has several tasks to perform at the same time, and can make several catastrophic errors without any advance warning. For instance, an important requirement in damage control is requesting permission before securing a valve. As part of the story development in FCT, the student is aware that there is a fire being fought on elsewhere on the ship. If the student attempts to repair a leaking fire main by securing the valve, a shipmate elsewhere on the ship gets injured when the water to his hose is cut-off and the fire he was fighting goes out of control. The importance of this consequence is emphasized using a short cut-scene shown from the perspective of the shipmate fighting the fire (see Figure 6).

EVALUATION

Ranking	Overall reaction to game	The game was stimulating	It was easy to play the game	The instructions were clear
0	0%	2%	0%	0%
1	0%	2%	2%	0%
2	2%	0%	0%	0%
3	0%	2%	0%	0%
4	3%	2%	0%	0%
5	6%	11%	3%	4%
6	15%	7%	6%	3%
7	25%	27%	13%	12%
8	25%	18%	16%	19%
9	25%	30%	60%	62%

Table 4: Overall Usability Responses
(0=low rating, 9=high rating)

The effectiveness of the Flooding Control Trainer was evaluated in two key studies -a usability study and a validation study -conducted with students from RTC. Prior to each study, a pilot was held. Following each study, the FCT was enhanced based on feedback.

Usability Study

A usability study was conducted in October 2008 with seventy subjects. The subjects were drawn from a population of recent graduates from RTC who had not yet deployed to their first posting (i.e., they had completed Battle Stations 21). The vast majority (92%) of respondents described their comfort with computers as average or above. Participation was voluntary. Each participant had approximately two hours available to play the Flooding Control Trainer. Performance was observed and rated by several trained observers during gameplay. Following gameplay, the subjects completed customized versions of two usability forms: the Questionnaire for User Interface Satisfaction (QUIS) (Chin et al., 1988) and the System Usability Scale (Brooke, 1996; Copyright Digital Equipment Corporation, 1986). Usability results were very positive; most subjects rated the game as 7 or higher on a scale of 0 to 9, with 9 being a strong positive rating (see Table 4). There were no differences associated with any background variable.

For the usability study, the FCT only had two missions -a tutorial and a flooding mission requiring a fire main pipe to be patched. During the usability study, we noticed that the students were having trouble navigating around the ship and introduced a condition in which one group of subjects was given a short verbal refresher on interpreting compartment identifiers and a reference sheet to use while playing. While the treatment group showed no differences in the QUIS items, they made significantly fewer navigation errors and were less likely to fail (Bowers et al., 2009). In response to this result, we identified the need for an additional training level focusing on navigation. This became a level between the tutorial and flooding missions.

Validation Study

In April 2009, a validation study was conducted to test the benefits of the Flooding Control Trainer (FCT) on individual performance within a flooding control test scenario in the Battle Stations 21 environment. Thirty-one recruits participated in the study. These recruits had completed RTC training but had not yet done the BS21 capstone evaluation. Sixteen of the participants formed the control group, and fifteen formed the treatment group. The treatment group played the FCT for one hour and then took the test scenario two days later. The control group had no extra training and took the same test scenario.

In the test scenario, an individual recruit was given orders to report to DCC by a primary facilitator. DCC (played by another RTC facilitator) ordered them to dress out and report to a specific location to investigate potential flooding. The recruit needed to perform the appropriate actions, find the compartment and communicate appropriately. At the indicated compartment, the recruit needed to safely investigate the compartment for flooding, report the situation and, upon receiving orders, patch a leaking pipe with a jubilee patch. The recruit needed to perform all their tasks with no help from the facilitators.

The recruits were assessed on a number of behaviors related to communications, decision-making, situational awareness and navigation within the ship. Performance differences between the groups were striking. Decision making errors were reduced by 50%, Communication errors were reduced by up to 80%. Situational Awareness and Navigation skills were improved by 50%.

	Control	*Treatment*
Entered the flooding compartment without appropriate PPE	*67%*	*28%*
Identified themselves on first contact with damage control	*7%*	*93%*
Repeated back commands from DCC	*7%*	*57%*
Described the leak correctly	*16%*	*36%*
Went to the wrong deck	*33%*	*0%*

Table 5: Performance Differences between Control Group and Treatment (Game) Group

A full description of the validation study and results will be presented elsewhere (in preparation). Differences on some of the key behaviors are given in Table 5. The treatment group performed significantly better in each case.

In addition to these specific measures, the behaviors of the two groups were visibly quite different in terms of their stress level and independence. The individuals in the control group generally appeared confused as to what they should

do and made frequent requests for help. The individuals in the treatment group were generally confident in their actions, made few requests for help and appeared to be enjoying the challenge of the test.

CONCLUSIONS

By working actively to weave together the instructional, narrative, gaming and assessment elements into a cohesive whole, we developed a flooding control training game for the U.S. Navy that demonstrates significant learning benefits and transfer of learned skills. Given the strong benefit of the Flooding Control Trainer game for improving communication, decision-making, situational awareness and navigation skills in individuals, we are confident that the game will have a strong effect on team performance within the Battle Stations 21 capstone evaluation. We predict that trainees who train using the game prior to BS21 will demonstrate significant improvements in the skills that are practiced directly in the game, higher-order skills that can become the focus of the trainee's attention, and overall performance due to higher confidence in their ability to cope with the challenge. Further, though the game provides practice on skills needed in the BS21 flooding control scenario, many of the skills reinforced in the game are relevant to a number of additional BS21 scenarios as well. Thus, we predict that the trainees who train using the game will show improvements across a variety of BS21 scenarios, not just the flooding scenario. The FCT is currently in the process of being deployed at RTC, and we hope to have additional results on the effectiveness of our system before the end of the year.

We are currently in the second year of a three year effort and have several enhancements to our trainer planned. We are currently enhancing the FCT with an additional level that includes a complex flooding situation requiring a higher degree of prioritization and more complex safety protocols. The additional level will provide a strong challenge for recruits, and will bring the Flooding Control Trainer to a level of complexity that begins to address the training needs of the Navy's technical schools.

We are also currently extending our training system to train fire fighting skills. As with flooding control, a suite of several missions of increasing complexity are planned. In our final year of effort, we plan to create several scenarios addressing skills required in combating a mass casualty situation. Together, these three domains will provide a solid foundation in shipboard damage control that is suitable for recruits and for sailors at technical schools.

Our long-term vision is that games such as the FCT will form a regular part of the training that occurs in the schoolhouses and in the fleet. Damage control is a critical skill set required of all sailors, and one that must be kept fresh throughout a sailor's career. We believe it is an ideal domain for demonstrating the utility and effectiveness of game-based training for the Navy.

REFERENCES

Bandura, A. (1977). Self-efficacy: Toward a unifying theory of behavioral change. *Psychological Review, 84, 191-215*.

Bandura, A. (1986). *Social foundations of thought and action: A social cognitive theory*. Englewood Cliffs, NJ: Prentice-Hall.

Bowers, C., Cannon-Bowers, J. & Hussain, T. (2009). Considering user knowledge in the evaluation of training system usability, *Human-Computer Interaction International* 2009.

Bransford, J. D., Brown, A. L., & Cocking, R. R. (Eds.) (1999). *How people learn: Brain, mind, experience, and school.* Washington, DC: National Academy Press.

Bransford, J. D., Sherwood, R. D., Hasselbring, T. S., Kinzer, C. K., & Williams, S. M. (1990). Anchored instruction: Why we need it and how technology can help. In D. Nix & R. J. Spiro (Eds.), *Cognition, education, and multimedia: Exploring ideas in high technology* (pp. 115-141). Hillsdale, NJ: Lawrence Erlbaum Associates.

Brooke, J. (1996). SUS: a "quick and dirty" usability scale. In P. W. Jordan, B. Thomas, B. A. Weerdmeester & A. L. McClelland (eds.) *Usability Evaluation in Industry.* London: Taylor and Francis.

Cannon-Bowers, J., Bowers, C., & Sanchez, A. (2008). Using Synthetic Learning Environments to train teams. *Work group learning: Understanding, improving and assessing how groups learn in organizations* (pp. 315-346). New York, NY : Taylor & Francis Group/Lawrence Erlbaum Associates.

Cartier, J. L., & Stewart, J. (2000). Teaching the nature of inquiry: Further development in a high school genetics curriculum. *Science and Education, 9,* 247-267.

Chi, M. T. H. (2000). Self-explaining: The dual processes of generating inference and repairing mental models. In R. Glaser (Ed), *Advances in instructional psychology: Educational design and cognitive science, Vol. 5*. (pp. 161-238). Mahwah, NJ: Lawrence Erlbaum Associates.

Chi, M. T. H., Glaser, R., & Farr, M. J. (Eds.). (1988). *The nature of expertise*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Chin, J. P., Diehl, V. A., & Norman, K. L. (1988). Development of an instrument measuring user satisfaction of the human-computer interface. *Proceedings of ACM CHI'88 Conference on Human Factors in Computing Systems*, 213-218.

Cognition and Technology Group at Vanderbilt (CGTV) (1992). Anchored instruction in science and mathematics: Theoretical basis, developmental projects, and initial research findings. In R. A. Duschl & R. J. Hamilton (Eds.), *Philosophy of science, cognitive psychology, and educational theory and practice. SUNY series in science education* (pp. 244-273). Albany, NY: State University of New York Press.

Cognition and Technology Group at Vanderbilt (CGTV) (1997). *The Jasper Project: Lessons in curriculum, instruction, assessment, and professional development.* Mahwah, NJ: Lawrence Erlbaum Associates.

Cognition and Technology Group at Vanderbilt (CTGV) (2000). Adventures in anchored instruction: Lessons from beyond the ivory tower. In R. Glaser (Ed), *Advances in instructional psychology: Educational design and cognitive science, Vol. 5.* (pp. 35-99). Mahwah, NJ: Lawrence Erlbaum Associates.

Darken, R., McDowell, P., Johnson, E. (2005). The Delta3D open source game engine. *IEEE Computer Graphics and Applications,* 25(3). Los Alamitos, CA: IEEE Computer Society Press, 10-12.

Diller, D., Roberts, B., Blankenship, S., & Nielsen, D. (2004). "DARWARS Ambush! – Authoring lessons learned in a training game," *Proceedings of the 2004 Interservice/Industry Training, Simulation and Education Conference (I/ITSEC),* Orlando, FL, December.

Glaser, R. (1989). Expertise in learning: How do we think about instructional processes now that we have discovered knowledge structure? In D. Klahr & D. Kotosfky (Eds.), *Complex information processing: The impact of Herbert A. Simon* (pp. 269-282). Hillsdale, NJ: Lawrence Erlbaum Associates.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science,* 7, 155-170.

Gist, M. E., Schwoerer, C., & Rosen, B. (1989). Effects of alternative training methods on self-efficacy and performance in computer software training. *Journal of Applied Psychology,* 74, 884-891.

REFERENCES CONTINUED

Gist, M. E., Stevens, C. K., & Bavetta, A. G. (1991). Effects of self-efficacy and post-training intervention on the acquisition and maintenance of complex interpersonal skills. *Personnel Psychology*, 44, 837-861.

Goldstein, I. (1993). *Training in organizations: Needs assessment, development, and evaluation* (3rd ed.). Belmont, CA US: Thomson Brooks/Cole Publishing Co.

Gunter, G. A., Kenny, R. F., & Vick, E. H. (2006). A case for a formal design paradigm for serious games. *The Journal of the International Digital Media and Arts Association*, 3, 93-105.

HPC (Human Performance Center), Naval Service Training Command Detachment (2008). *Battle Stations 21 Performance Measurement Execution Analysis Report,* prepared on February 15, 2008 for the Commander, Naval Service Training Command (NSTC).

Hussain, T. S., & Ferguson, W. (2005). Efficient development of large-scale military training environments using a multi-player game, 2005 *Fall Simulation Interoperability Workshop*, p. 421-431.

Hussain, T. & Feurzeig, W. (2008). Methods and tools for the development of effective training games, *Session on Computer Games and Team and Individual Learning at 2008 American Educational Research Association (AERA) Annual Meeting.*

Hussain, T., Feurzeig, W., Cannon-Bowers, J., Coleman, S., Koenig, A., Lee, J., Menaker, E., Moffitt, K., Murphy, C., Pounds, K., Roberts, B., Seip, J., Souders, V. and Wainess, R. (in press) Development of game-based training systems: Lessons learned in an inter-disciplinary field in the making, in J. Cannon-Bowers & C. Bowers (Eds.) *Serious Game Design and Development: Technologies for Training and Learning*

Hussain, T. S., Weil, S. A., Brunye, T., Sidman, J., Ferguson, W., & Alexander, A. L. (2008). Eliciting and evaluating teamwork within a multi-player game-based training environment in H.F. O'Neil and R.S. Perez (Eds.), *Computer Games and Team and Individual Learning* (pp. 77-104). Amsterdam, The Netherlands: Elsevier.

Johnson, W. L., Wang, N., & Wu, S., (2007). "Experience with serious games for learning foreign languages and cultures," *Proceedings of the SimTecT Conference*. Australia.

Kirlik, A., Fisk, A., Walker, N., & Rothrock, L. (1998). Feedback augmentation and part-task practice in training dynamic decision-making skills. *Making decisions under stress: Implications for individual and team training* (pp. 91-113). Washington, DC US: American Psychological Association

Locke, E. A., Shaw, K. N., & Saari, L. M. (1981) Goal setting and task performance: 1969-1980. *Psychological Bulletin*, 90, 125-152

Locke, E. A., & Latham, G. P. (1990). Work motivation: The high performance cycle. In U. Kleinbeck & H. Quast, (Eds.), *Work motivation* (pp. 3-25). Hillsdale, NJ: Lawrence Erlbaum Associates.

Mayer, R. E. (2001). *Multimedia learning*. Cambridge, England: Cambridge University Press.

McDonough, J. P., & Dmochowski, N. (2008). Deployable Virtual Training Environment. Presentation at the 2008 Defense GameTech Users' Conference, Orlando, FL. Retrieved June 29, 2009, from **http://www.peostri.army.mil/PAO/pressrelease/FIL ES/GameTech08/04302008/Marine%20Corps%20 Status%20VBS2_Dmochowski.pdf**

Murphy, C., Rodgers, C., Guthrie, D. & McDowell, P. (2008). Building training games with the Delta3D Simulation Core. Tutorial given at the 2008 *Interservice/Industry Training, Simulation and Education Conference.* Retrieved June 29, 2009, from **http://www.delta3d.org/filemgmt_data/files/Tutoria l_629_v1.1.pdf.**

REFERENCES CONTINUED

O'Bea, M., & Beacham, J. (2008). How do You Train When Your Equipment is 7000 Miles Away? *Proceedings of ITEC 2008,* Stockholm, Sweden.

O'Neil, H. F., & Perez, R. S. (2008). *Computer Games and Team and Individual Learning.* Amsterdam, The Netherlands: Elsevier.

O'Neil, H. F., Wainess, R., & Baker, E. L. (2005). Classification of learning outcomes: Evidence from the computer games literature, *The Curriculum Journal, 16 (4)*. 455-474.

Raghavan, K, Satoris, M. L., & Glaser, R. (1997). The impact of model-centered instruction on student learning: The area and volume units. Journal of *Computers in Mathematics and Science Teaching, 16,* 363-404.

Raghavan, K, Satoris, M. L., & Glaser, R. (1998). The impact of MARS curriculum: The mass unit. *Science Education, 82,* 53-91.

Roberts, B., Diller, D., & Schmitt, D. (2006). Factors affecting the adoption of a training game, *Proceedings of the 2006 Interservice/Industry Training, Simulation and Education Conference.*

Roman, P. A., & Brown, D. (2008). Games -Just how serious are they? *Proceedings of the 2008 Interservice/Industry Training, Simulation and Education Conference (I/ITSEC),* Orlando, FL, December.

Rothman, F., & Narum, J. L. (1999). Then, Now, and In the Next Decade: A Commentary on Strengthening Undergraduate Science, Mathematics, Engineering and Technology Education. Project Kaleidoscope, Washington, DC.

Salas, E., & Cannon-Bowers, J. A. (2000). The anatomy of team training. In S. Tobias & J. D. Fletcher (Eds.), T*raining and retraining: A handbook for business, industry, government, and the military* (pp. 312-335). New York, NY: Macmillan.

Salas, E. & Cannon-Bowers, J.A. (2001). The science of training: A decade of progress. *Annual Review of Psychology 52:* 471-499.

Schunk, D. H., & Ertmer, P. A. (1999). Self-Regulatory Processes During Computer Skill Acquisition: Goal and Self-Evaluative Influences. *Journal of Educational Psychology,* 91, 251-260.

Smith, R. D. (2008). Five forces driving game technology adoption, *Proceedings of the 2008 Interservice/Industry Training, Simulation and Education Conference.*

Stewart, J., Cartier, J. L., & Passmore, C. M. (2005). Developing an understanding through model-based inquiry. In M.S. Donovan & J.D. Bransford (Eds.), *How Student Learn: History, Mathematics and Science Inquiry in the Classroom,* (pp. 515-565). Washington, DC: National Academies Press.

Vogel, J., D. S., Cannon-Bowers, J. A., Bowers, C. A., Muse, K., & Wright, M. (2006). Computer gaming and interactive simulations for learning: a meta-analysis. *Journal of Educational Computing Research, 34(3)*, 229-243.

Wilson, K. A., Bedwell, W. L., Lazzara, E. H., Salas, E., Burke, C.S., Estock, J. L., Orvis, K. L., & Conkey, C. (2009) Relationships between game attributes and learning outcomes: Review and research proposals, *Simulation Gaming 2009, 40(2),* 217-266.

Wurtele, S. (1986). Self-efficacy and athletic performance: A review. *Journal of Social & Clinical Psychology, 4(3),* 290-301.

Zimmerman, C, Raghavan, K., & Sartoris, M. L. (2003). The impact of MARS curriculum on students' ability to coordinate theory and evidence. International *Journal of Science Education, 25,* 1247-1271.

This article was adapted from I/ITSEC 2009 and is reprinted with permission.

SCALABLE AND EMBEDDABLE DATA LOGGING FOR LIVE, VIRTUAL AND CONSTRUCTIVE SIMULATION

HLA, Link 16, DIS and more

Björn Möller
Pitch Technologies, Sweden
bjorn.moller@pitch.se

Fredrik Antelius
Pitch Technologies, Sweden
fredrik.antelius@pitch.se

Tom van den Berg
TNO, The Netherlands
tom.vandenberg@tno.nl

Roger Jansen
TNO, The Netherlands
roger.jansen@tno.nl

ABSTRACT

One of the most important simulation assets is the data that is collected during executions. Imagine being able to look back, analyze and reuse the data of simulations that have been run during the last decade. However, data logging has a number of challenges, not the least in today's environment where we need to train jointly and combined and mix a number of Live, Virtual and Constructive simulators, using different standards.

This paper summarizes some requirements for Live, Virtual Constructive (LVC) data logging as well as replay. It also describes some early experiences from developing and testing a data logger that can perform fully synchronized, simultaneous data logging of High Level Architecture (HLA), Distributive Interactive Simulation (DIS), Link 16 and other data streams. Some details are given on aspects like embedding, chase play, ownership and import/export. Some challenges and limitations when mixing these different interoperability and data link standards are also covered.

KEYWORDS:

Simulation, Training, After Action Review, Interoperability, Data logging, HLA, DIS, Link 16, Voice, Viking

AUTHOR BIOGRAPHIES

BJÖRN MÖLLER is the vice president and co-founder of Pitch, the leading supplier of tools for HLA 1516 and HLA 1.3. He leads the strategic development of Pitch HLA products. He serves on several HLA standards and working groups and has a wide international contact network in simulation interoperability. He has twenty years of experience in high-tech R&D companies, with an international profile in areas such as modeling and simulation, artificial intelligence and Web-based collaboration. Björn Möller holds an M.Sc. in Computer Science and Technology after studies at Linköping University, Sweden, and Imperial College, London. He is currently serving as the vice chairman of the SISO HLA Evolved Product Support Group.

FREDRIK ANTELIUS is a Lead Developer at Pitch and is a major contributor to several commercial HLA products. He holds an M.Sc. in Computer Science and Technology from Linköping University, Sweden.

TOM VAN DEN BERG is scientist in the Modeling, Simulation and Gaming department at TNO, The Netherlands. He holds an M.Sc. degree in Mathematics and Computing Science from Delft Technical University. His research area includes simulation systems engineering, distributed simulation architectures and concept development & experimentation.

ROGER JANSEN is a member of the scientific staff in the Modeling, Simulation and Gaming department at TNO, The Netherlands. He holds an M.Sc. degree in Computing Science and a Master of Technological Design (MTD) degree in Software Technology, both from Eindhoven University of Technology, The Netherlands. He works in the field of distributed simulation and his research interests include distributed computing and simulation interoperability.

1. INTRODUCTION

Data is one of the most important results or outputs of computer-based modeling and simulation. Even though computer-based modeling and simulation is a relatively

young discipline, many models have been executed over the years, a lot of data has been produced and most of this output data is forever lost, in many cases since it was not logged.

No matter if a simulation is executed in real-time or using logical time, time-stamped simulation data can be logged for later use, like analysis or after-action review. This data will typically have a closer connection to the original set of simulators that produced the data than what a naïve user may initially think. It will usually be necessary to understand the goal, the assumptions and the limitations of the original simulators and scenarios to be able to play it back and use in a meaningful way. Nevertheless, the simulation data can be highly useful, both for the original purpose and for new purposes, such as input to other simulations, testing, training, and for new types of analysis.

1.1 An LVC perspective on data logging

Taking a Live-Virtual-Constructive [1] perspective on data logging adds a number of additional aspects to the above, for example (see Figure 1):

- A challenging mix of simulation standards and protocols may need to be supported, usually together with a set of corresponding information exchange data models ("FOMs") that may be more or less coherent. More widely used types of data to be recorded include HLA [2], often with the RPR FOM [3], DIS [4] and voice (both as part of the HLA/DIS communication and using other ways of communication). Additional types of data may include Link 16 [5], Test and Training Enabling Architecture (TENA) [6], streaming video, and proprietary protocols, for example for Command and Control systems.
- In a virtual or constructive model the data values in each model may define the ground truth. In a live simulation we can only attempt to capture measurements or perceived truth. One example of this is positions measured using a Global Positioning System (GPS) where the inaccuracy may be measured in meters and will vary over time. The time stamps for data from different live sources may also need to be adjusted when data from different sources is merged.
- While a constructive or virtual simulation can be re-run, you may only be given one, or a very limited number of opportunities to capture output data from a live simulation. One example of this is the firing of a prototype missile in a test range.
- Live simulations may require wireless data connections to some of the players. This may result in less reliable communication lines, leading to gaps in logged data. Additional precautions may need to be considered to address this problem.

In many cases there may be no major differences between the collection of data from a real life system for LVC simulation purposes and for other purposes.

Still, the ability to combine data from several Live-Virtual-Constructive sources makes the potential of this data even higher. It allows us to understand a bigger picture than before, to train in a more realistic and effective way, and to better analyze the total impact of new concepts.

1.2 The role of the data logger in a simulation

A data logger is typically a software application that is either built-in into a simulation application or that is stand-alone. It may connect to one or more applications using a network protocol like DIS or interoperability services, like HLA. A data logger for HLA or DIS is usually more reusable than a built-in proprietary data logger but it may be limited to recording the public data provided in the FOM or the DIS protocol.

The most common functionality of a data logger includes:

- Recording of time stamped data from a data source like HLA or DIS into a file or a database.
- Playing back all or selected parts of the recorded data to a data sink of the same type (HLA or DIS). This may be done at the original speed, scaled to lower or higher speed, or using a completely different time-advance pattern, for example using HLA Time Management and/or event driven time advance.
- Support for human inspection of the data in a user interface.
- Support for automated inspection and analysis of the data through an API.
- Making the recorded data available in other formats like databases and plain text formats.
- Managing the timeline, for example by setting bookmarks or moving the playback time to a bookmark or a specific time value.
- Filtering the data during recording or playback.
- Adapting the data during playback, for example DIS exercise id, DIS entity id or HLA object instance name.

2. USE CASES FOR LVC DATA LOGGING

There are many ways to benefit from data logging in LVC simulations. Some of the more common applications are described here. The use cases are based on the experiences from a number of simulation systems developed within TNO as well as practical experiences provided by staff at Pitch.

2.1 Simulation for training

Simulation for training is a common application where data logging is used. One particular class of training applications is the virtual, man-in-the-loop simulations, for example for training pilots, drivers, forward air controllers or straddle carrier operators. Figure 2 is an example of a virtual simulator for the training of Forward Air Controllers.

Figure 2: TNO Forward Air Controller simulator.

Data logging in these simulations is mainly used to record and playback an exercise in real time, where instructors use VCR type functions to control data logging and playback.

The characteristics of these simulations are:

- The simulation is typically Virtual.
- It is a real time simulation where HLA Time Management is not used.
- Data logging is used for simulation data (ground truth), and sometimes also live voice or video data.
- The execution is controlled via start, stop, pause, and resume management messages.
- Simulation applications may join or leave the simulation execution when they want.
- Real-time replay of the logged data is used for after action review.

Since these simulations have been around for a while, the required functionality for recording and replay is generally well understood.

From a high-level point of view, three major states can be identified for this kind of simulation.

- Preparation. In this state a training scenario is prepared and previously recorded data may be used for the construction of a new scenario. Common simulator functions in this state are: create new scenario, edit scenario, delete scenario, load scenario and save scenario. Editing a scenario involves many different functions which will differ per training application, such as entity placement on a 2D map, route planning and entity behavior configuration. When the scenario is started, the prepared scenario becomes the initial situation at the start of the scenario execution.
- Execution. In this state the scenario is executed over time. Simulation data and other relevant data are recorded for after action review. It is possible to bookmark certain events for use in after action review. When the execution is stopped, the existing situation may become a new scenario in the preparation state.
- After Action Review. In this state a previously recorded exercise can be replayed and visualized in the original simulators or in 3D or 2D viewers. It is possible to view the list of available bookmarks, to jump to a bookmark or to a certain point in time in the recording. It is also possible to pause and resume the replay. When the after action review is stopped, the existing situation may become a new scenario in the preparation state.

2.2 Simulation for analysis

There are many different types of analysis models. Here we have chosen to focus on stochastic simulation (Monte Carlo [7] simulation). Stochastic simulation typically involves thousands or more simulation runs, varying one or more parameters. The simulation runs can be long lasting (in elapsed time), and are executed in non-real time. In most cases these simulations run as-fast-as-possible. Analysis involves processing and aggregating large amounts of data that has been recorded over the various runs. Ad-hoc queries on the recorded data may be needed to zoom in on

certain aspects. Analysis is usually performed afterwards when all the data can be aggregated and searched.

Two examples where stochastic simulation is applied are described in earlier papers [8][9]. In [8] the effect of dynamic train management is studied, using small stochastic variations in the train schedule. Figure 3 shows the study area for dynamic train management, where the Dieze Bridge forms a bottleneck for trains. In [9] a footprint analysis is performed to determine the region that a ship can defend against a missile, using stochastic variations in sensor behavior.

In both examples a large amount of data is collected during the simulation execution and transferred to a dedicated analysis application. Stochastic simulation also requires more extensive simulation states to control simulation execution, such as states for simulation initialization, warm-up, steady state execution, iterations and shutdown. This is quite different from the relatively simple simulation states in the training case.

We can summarize the characteristics of a stochastic simulation as follows:

- The simulation is typically Constructive.
- It is a non-real time simulation where HLA Time Management is used.
- Data logging is used for simulation data (ground truth) as well as simulated operational data, like the Link 16 BOM (perceived truth).
- Execution is controlled via synchronization points and save/restore points
- All applications need to be present throughout the simulation execution.

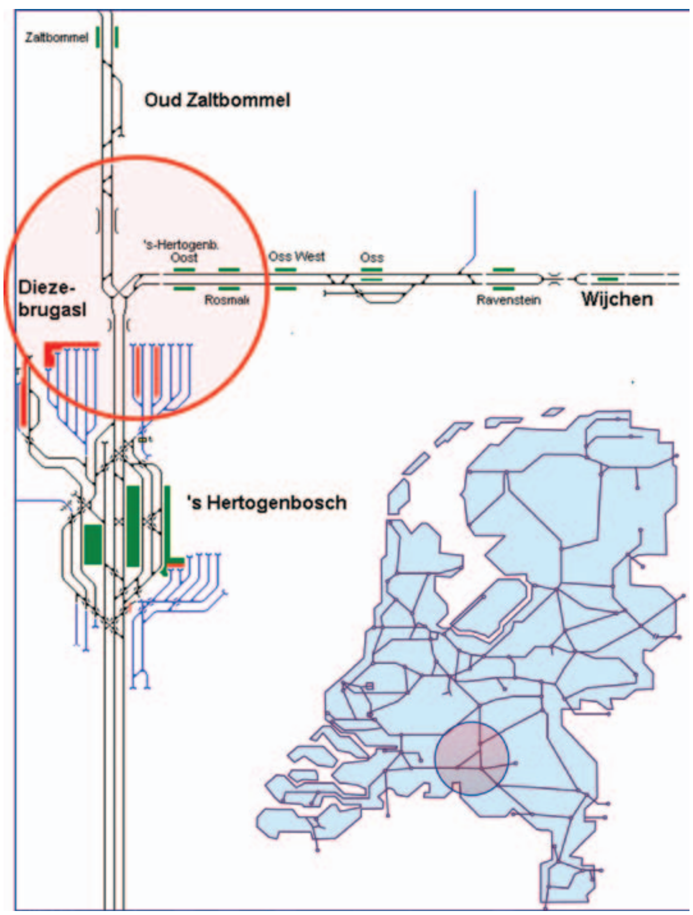

Figure 3: Study area to analyze dynamic train management using Monte Carlo simulation [8].

- Replay of certain runs may be possible, but results may also just be charts such as bar or line charts of aggregated data

2.3 Simulation for test and evaluation of live systems

This use case involves connecting real-time (operational, live) systems to a simulation for test and evaluation. The idea behind this is to test and evaluate a system early in the development cycle and certainly before the system arrives in the target environment. A simulation can provide, for example, stimuli or 'ground truth input' in order to verify if the resulting behavior of the system is correct. Alternatively a data logger may be used to replay previously recorded data to stimulate a system. The resulting system behavior may be the transmittal of certain tactical (operational) messages, which may be fed back in the simulation for additional stimuli. Thus simulation for test and evaluation involves simulation data, operational data and real-time execution.

Analysis involves correlating simulation data with operational data to verify if the right data was generated at the right moment where timing of certain messages may be important. For example, which Link 16 track corresponds to which simulation entity? Are the correct tactical messages generated across all phases of a missile engagement? An example of this use case is shown in Figure 4 where JROADS (Joint Research On Air Defence Simulation) is connected with live systems using Link 16.

Analysis may be performed on-line (during simulation execution) or off-line (after simulation execution). With on-line analysis, both simulation data and operational data

are monitored during the simulation execution. It is possible to pause the monitoring in order to look at certain data, while at the same time the recording of data continues. The monitoring can be resumed and fast forwarded to catch up with the ongoing execution, so called chase play, just like modern hard-disk video recorders that can record and play a film at the same time, while jumping back and forth in the film. Bookmarking may be used to jump to certain important points that have been marked earlier in the recorded data.

With off-line analysis the recorded simulation data and operational data is reviewed after the execution has finished. Data may be replayed in real time or faster/slower than real time (n times real time). Important to note is that the timing of messages that are replayed can be important or even critical, due to the correlation between simulation data and operational data over time. Also, data from external sources may need to be combined with the recorded data, such as log files from command and control systems. Data from external sources can be provided in different formats (e.g., comma-separated value file or xml file). An application for off-line analysis is described in [10].

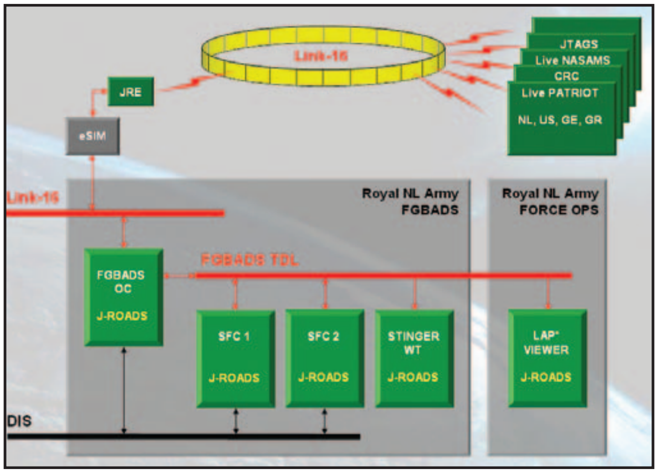

Figure 4: JROADS simulation integrated with live systems via a tactical data link.

Again, we can summarize the characteristics of a simulation for test and evaluation as follows:

- The simulation can be regarded as Live.
- It is a (hard) real-time simulation where HLA Time Management is not used.
- Data logging is used for simulation data (ground truth) as well as live/simulated operational data, like Link 16 (perceived truth).
- Execution is controlled via start/stop management messages, monitoring via pause/resume/jump messages.
- Depending on the system, all applications in the simulation environment need to be present throughout the simulation execution.
- Real time and non-real time replay of data is used for after action review.

2.4 Federation development

Logged simulation data is highly useful to minimize time, cost and risk during the development of simulation software, in particular when adding HLA or DIS interfaces. The output data of a simulator can be logged, inspected and checked against the expected output. Well-known, correct simulation data can be fed into a simulator from a data logger to check stability and correct behavior. You may even exchange logged data between several simulators before you connect them for real. An integration leader may apply a pre-integration methodology where all systems are required to be tested against a well defined set of test data before they are allowed to join the full federation. Data logging for simulator development is applicable to all the above types of simulation. It generally shares all of the above requirements but the requirement to be able to exchange data files is prominent.

3. REQUIREMENTS AND CHALLENGES FOR DATA LOGGERS

The different use cases all lead to a set of requirements for recording and replay. Ideally we're looking for a multi-purpose recording and replay capability that can fulfill all requirements. This section of the paper lists the requirements and maps them to the use cases above that are most relevant.

3.1 Data streams

Requirement 1: The data logger must support several data streams (HLA, DIS, etc, as required by the simulation), or be extendable with new data streams.

Most applicable to: Training, Test and Evaluation

Today's simulation environments are open and all kinds of systems can be connected, generating different types of

data. A well known example in the missile and air defense domain is Link 16. Another example is voice. Recording should not just be limited to simulation data.

3.2 Session management

Session management concerns the management of recording sessions: create a recording session with the required (DIS, HLA, etc) simulation connection parameters; destroy a previously created recording session; open a recording session for replay; close a previously opened recording session; start, stop, pause, resume the recording or replay within a session; jump to bookmark or jump to time within a session.

Requirement 2: The data logger must be able to record data streams and store them as a recording session. Recorded data streams (DIS, HLA, etc.) must be stored together in a recording session.

Most applicable to: All use cases

Requirement 3: The data logger must be able to retrieve a recording session and replay all or a subset of the recorded data streams.

Most applicable to: All use cases

Data streams in a recording session should also be replayed together. The precise timing of data stream messages may be important. For example if in data stream A messages are recorded at time 0, 5, 10, ..., and in data stream B at time 2, 5, 8, ..., then these should also be replayed exactly this way. Thus, during replay, data streams in a recording session must remain synchronized in time.

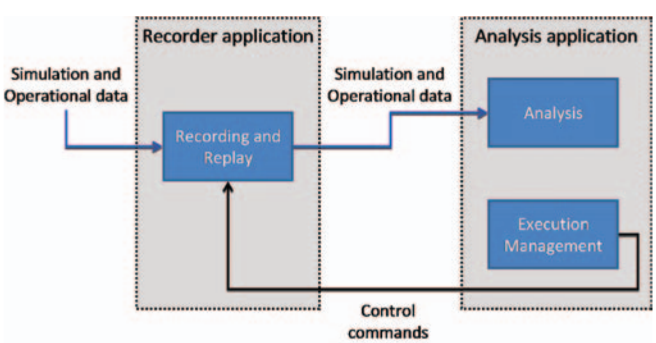

Figure 5: Activity diagram for concurrent recording and replay.

Requirement 4: The data logger must be able to replay a data stream in a different format than was recorded.

This requirement implies that the data logger is aware of the data being recorded. For example, record a DIS data stream and replay the DIS data stream as an XML formatted data stream.

Most applicable to: Test and Evaluation

Requirement 5: The data logger must be able to pause/resume/fast forward/fast backward a replayed recording session.

Most applicable to: All use cases

Requirement 6: The data logger must support the filtering of data from a data stream on recording and on replay.

Most applicable to: All use cases

Requirement 7: The data logger must support the concurrent recording and replay of a data stream.

Most applicable to: Test and Evaluation, Federation development

Usually replay happens only when recording has finished. But in some cases it must be possible to view and analyze data streams while they are being recorded. Thus, data streams are replayed at the same time as they are recorded (concurrently). Also the requirements to pause/resume/fast forward/fast backward, to jump to a bookmark or jump to a point in time, and replay in a different format apply on the replayed data streams. When a replayed data stream lags behind on the recording, it is called "chase play".

Figure 5 shows the principle.

The data streams that come out of the Recording and Replay activity should not be replayed on the same DIS exercise or HLA federation where the data is recorded from. Thus the data streams should be replayed in a different DIS exercise or HLA federation, or even in a different format, for example as XML on a TCP connection.

Requirement 8: The data logger must support the grouping of recording sessions and support the addition of meta-data to each group.

Most applicable to: Analysis

With Monte Carlo simulations, each run results in a recording session. Recording sessions of related runs (for example, where only the seed is different) should be grouped and have the variation number and other variable settings added as meta-data.

3.3 Bookmark management

Requirement 9: The data logger must support the management of bookmarks (create, delete, update bookmark; retrieve bookmarks).

Most applicable to: All use cases

Requirement 10: The data logger must be able to jump to a bookmark or jump to a point in time in a replayed recording session.

Most applicable to: All use cases

When jumping to a certain point in time (say time T), it may be necessary to scan the data stream backwards in time to build up a complete picture for time T. For example, with a DIS data stream the data logger may need to scan back up to 13 seconds in order to find all entity state updates for time T.

3.4 Time management

Requirement 11: The data logger must be able to record data in a real-time simulation (which does not use HLA Time Management or similar services).

Most applicable to: Training, Test and Evaluation, Federation development

Requirement 12: The data logger must be able to record data in a (real time or non-real time) HLA time—managed simulation.

Most applicable to: Analysis, Federation Development

Time management concerns the use of HLA Time Management services when the simulation is time-managed. With a time-managed simulation a data logger is usually a time constrained federate in recording mode and, depending on the federation, a time regulating federate in replay mode. Recording and Replay needs to support HLA Time Management.

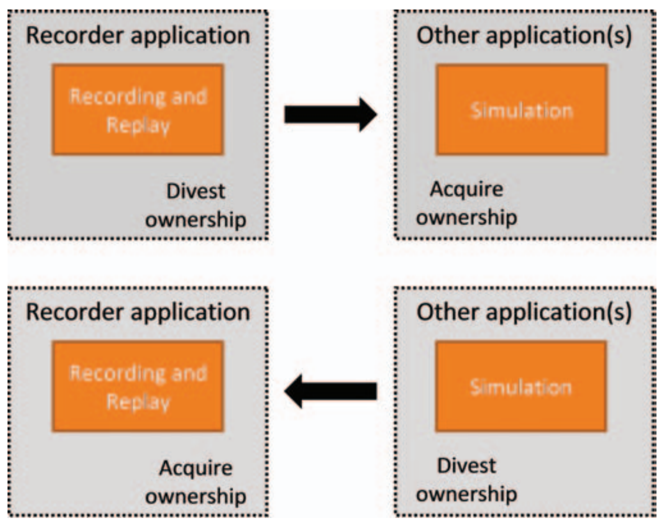

Figure 6: Ownership transfer on mode change: (1) from After Action Review to Execution or Preparation (top) and (2) from Execution or Preparation to After Action Review (bottom)

Requirement 13: The data logger must be able to replay a recording session at different speeds (real time or faster/slower than real time).

Most applicable to: Test and Evaluation

Restrictions may apply for certain data streams in certain situations. For example, a DIS data stream may in some cases only be replayed real-time, otherwise dead-reckoning models in applications like viewers may not work correctly.

3.5 Ownership management

Requirement 14: The data logger must be able to transfer ownership of object instances in a certain data stream on a mode change (between recording and replay).

Most applicable to: Training

Ownership management concerns a transfer of ownership of HLA object instances when the mode changes between recording and replay. In some situations an ownership transfer is required, for example when the state of the

object instances provided in replay mode is used as the initial state for a new simulation execution. The data logger needs to release ownership and the different applications that model the object instances must acquire ownership. This is summarized in Figure 6.

3.6 Data management

Requirement 15: The data logger must support the exchange (import/export) of recorded data with other applications.

Most applicable to: Analysis, Test and Evaluation, Federation development

For export, there are different options to consider, for example: raw export (export the data streams as they were recorded), structured export (export the data streams to a structured format, e.g., a SQL database where the schema matches the HLA FOM).

3.7 Control and embedding

Requirement 16: The data logger must be able to handle execution management messages that are received via a data stream.

Most applicable to: Analysis

In some cases execution management messages (such as HLA synchronization points, HLA Save/Restore, and user defined simulation management interactions) need to be interpreted by the Recording and Replay activity. This can, for example, be a certain HLA interaction that identifies the end of a Monte Carlo simulation run. The Recording and Replay activity must provide hooks to handle these execution management messages. A default hook could implement some default behavior, like achieving an HLA synchronization point.

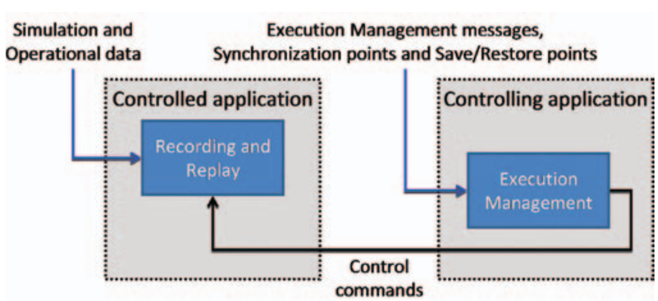

Figure 7: Activity diagram for recording mode

Requirement 17: The data logger must be embeddable in and completely controllable by another application, concerning all the earlier mentioned requirements.

Most applicable to: Training, Analysis, Test and Evaluation

This is an important requirement and allows recording and replay to be integrated with virtually any simulation application. Figure 7 shows an example of embedded control.

In this example the controlling application performs the activity execution management. It controls the application (i.e., the data logger) that performs the activity recording and replay. The controlling application handles, for example, the HLA synchronization points, HLA Save/Restore and Execution Management messages (such as start-resume and stop-freeze DIS PDUs in a DIS exercise) and, if needed, initiates mode changes on the controlled application. The controlled application (i.e., the data logger) does not interpret any Execution Management messages (these messages are just recorded as any other data) and achieves (by definition) any HLA synchronization or HLA save/restore in which it is involved.

Thus, with embedding, recording and replay is dedicated to performing just this activity while it is part of some application.

3.8 Scalability

Requirement 18: For initial testing, the data logger shall be able to operate on a regular computer without extensive setup. When used with a full federation, the data logger must be able to record/replay many different data streams concurrently and support long lasting and large recording sessions with tens of thousands of recorded events per second.

Most applicable to: All use cases

Note that there are advanced use cases where several data loggers could be used concurrently, for optimum

scalability, or in different locations to conserve bandwidth. Merging of the logged data may introduce additional challenges that are not covered in this paper.

4. PRACTICAL EXPERIENCES

This section summarizes our experiences from extending a COTS data logger with an additional LVC protocol.

4.1 About Pitch Recorder

Pitch Recorder, a COTS product, is a general purpose data logger with a rich set of features [11] targeted at LVC simulations. It provides parallel, synchronized recording of the following data streams:

- HLA data for any FOM with support for HLA 1.3, 1516-2000 and 1516-2010 RTIs.
- DIS version 4, 5 and 6 plus experimental PDUs
- Audio (for example for voice recording).
- User defined data streams, for example national C2 protocols

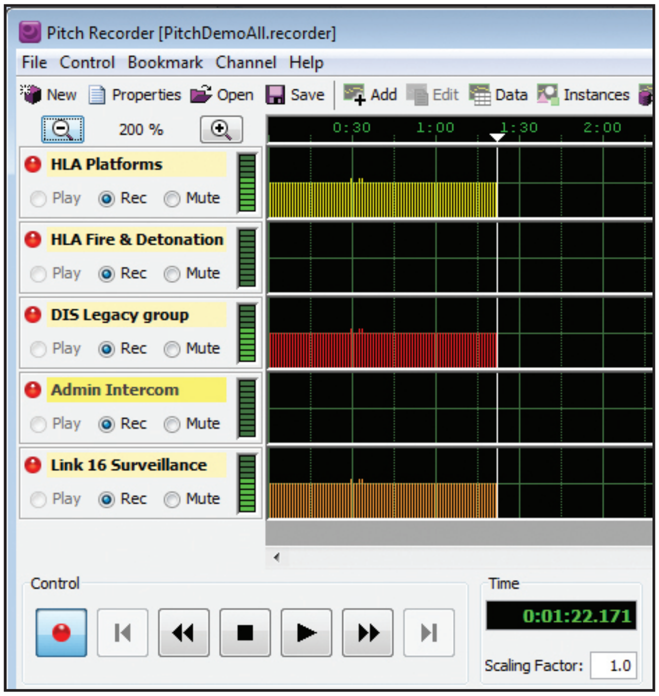

Figure 8: Channels in Pitch Recorder

In addition to the concept of a data stream, the Pitch Recorder introduces the concept of channels (see Figure 8). For an HLA data stream it is possible to configure different channels, for example, for land, sea and air entities as well as fire, detonation and radio. Pitch Recorder is not locked to any particular FOM and has been used for military, security, space, and civilian federations.

All data streams can be recorded, played back, filtered, inspected and exported to other programs. Complete recordings can also be exported to a package that can be sent by e-mail or other file transfer methods. Pitch Recorder can be used stand-alone or be embedded into a solution and externally controlled by another software application.

One of the more recent features of this product is a plug-in framework that allows the addition of new kinds of data streams for recording and replay.

4.2 Scalability Experiences

Pitch Recorder can record to small local databases for modest data flows. For large federations, high end COTS databases on dedicated hosts can be used for sustained logging of tens of thousands updates per second. Typical performance for Pitch Recorder in a lab test is more than 25 000 recorded HLA updates per second on a regular desktop computer.

An interesting scalability experience from a real training application is the recent Viking 11 exercise [12]. This exercise was described in ITEC 2011 keynote as the world's premier comprehensive exercise, including civilian, military and police participants. The exercise covered the planning and execution of a UN mandated Chapter VII Peace Operation/Crisis Response Operation. On the civilian side approximately 35 Non-Governmental Organizations participated. It was based on a scenario called Bogaland that contains a large number of challenges, for example, piracy, irregular forces, refugees, children in armed conflicts and reconstruction. Approximately 2500 persons from 31 nations were involved, participating from 9 different sites.

Examples of participating systems were JCATS, ICC, Sitaware, Exonaut, TYR, ASCOT and VBS2. The information exchange was based on an HLA Evolved infrastructure using Pitch pRTI Evolved version 4.2.5. Data was logged using Pitch Recorder with a separate database host running MySQL, saving data to a RAID-5 disk set. More than 160 hours of exercise were recorded amounting to more than 210 GB of data. The majority of this data was position updates. Note that the data rate varies a lot over time, with

a typical "idle rate" of 5000 updates per second. Voice data was also recorded using a separate Pitch Recorder since voice was handled in a separate federation.

One conclusion from this exercise is that it is important to fully understand how to configure the database manager (in this case MySQL) in order to guarantee that the data base sessions do not time out. Another, more obvious conclusion is the importance of powerful hardware to avoid overload during busy periods of the exercise.

4.3 Experiences from adding Link 16 support

As an engineering feasibility demonstration, a new data stream for Link 16 [5] recording and replay was added to Pitch Recorder by TNO. Tactical Data Link traffic like Link 16 is often emulated in simulation environments. Several protocols and wrappers are being used to provide the exchange of Link 16 messages between federates. The Standard Interface for Multiple Platform Link Evaluation (SIMPLE) [13] is widely supported and was selected for the engineering feasibility demonstration. The Link 16 data stream was added relatively easily to the Pitch Recorder, given that a Link 16 software library for receiving and sending Link 16 messages from/to a SIMPLE network was already available. A screenshot of the Link 16 plug-in is shown in Figure 9.

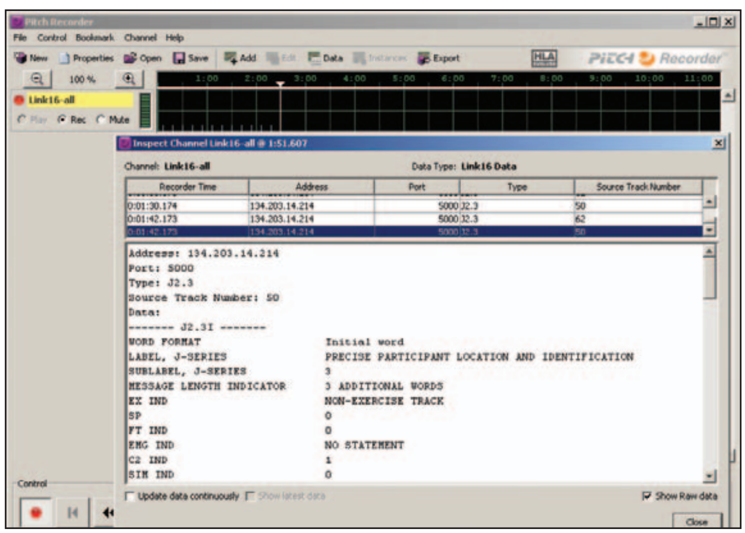

Figure 9: Screenshot of Pitch Recorder Link 16 recording.

The plug-in framework provides a set of Java interface classes that a plug-in must implement, for example, for sending and receiving data, and for providing a property window. Once the plug-in is constructed and compiled to a jar file, it is just a matter of dropping the jar file in the Pitch Recorder plug-in folder.

One of the reasons to choose SIMPLE Link 16 as a first candidate plug-in is to create the ability to record, replay and analyze DIS/HLA simulation data in combination with Link 16 tactical data. This data stream combination is often found in LVC air and missile defense simulation exercises, like JPOW (Joint Project Optic Windmill) [14].

The Link 16 plug-in for Pitch Recorder was successfully tested in the JROADS (Joint Research On Air Defence Simulation) simulation environment at TNO. JROADS is an extensive simulation tool to support air defense research and CD&E for the Netherlands armed forces. At JPOW, JROADS has been used for joint experimentation, analysis, and mission training for many years.

5. DISCUSSION

While generating requirements from the use cases, a number of challenges became obvious as to how these requirements should be implemented. This section summarizes some of them.

5.1 What data do we need to collect?

For many purposes, like after action review or analysis, there may be a requirement to use many types of data from the simulation. Some of them may be exchanged using HLA or DIS during the execution. Others may be internal variables in simulators or physical states of hardware. The challenge is how to collect the later type of data. Some approaches are to publish that data using HLA or to introduce a separate data stream for that data into the same or a different data logger. Using several data loggers creates problems when re-synchronizing the data. Sending additional data using HLA may only be practical for a limited set of data. Creating a specialized data stream for internal data from an application means a fair amount of work. The best approach has to be decided from case to case.

5.2 Data loggers and data awareness

One of the more difficult questions when designing a data logger is to what degree a data logger needs to be aware of the data it handles. Playing back data is usually more challenging than recording data and will sometimes require additional functionality in most participating simulators. Typical examples include:

- Handling of the life cycle of a simulated entity. This is handled differently in different architectures. If the playback of a DIS recording is paused and no data is sent for an aircraft for a certain time period, then listeners may delete that aircraft (unless all systems implement the freeze PDU). For HLA, a related problem is that a data logger may send out data for an aircraft that has not been created or that has attributes that are owned by another system.
- Handling of data where certain shared algorithms have been agreed. One example is dead reckoning where an aircraft has a certain speed that participating systems use for predicting its future position. When such data is played back at scaled time or even paused there is a risk that listeners may interpret the data in an unintended way.
- Handling of data that needs to be adapted. An example is the DIS exercise identification in a DIS data stream. The DIS exercise identification may be different on playback. Another example is time information. Time information may be adapted in order to replay data at another simulation time than it was recorded.

As can be seen from these examples a data logger may need to have deeper insights into both the simulation standard used and particular federation agreements.

5.3 Exchanging data that has been logged

It is likely that different organizations may want to use different data logging software. The same organization may even want to use different software over time or for different projects. Therefore, it would be of great value if different data loggers could exchange data using a standardized file format. While the internal format of a data logger may be optimized for fast search and execution, a data interchange format would be optimized for generality.

One strongly related topic is a long-term data archival format that ideally would be the same as a standardized data interchange format.

6. CONCLUSIONS

This paper has presented a number of use cases, requirements and challenges for data logging in an LVC environment. Although the different use cases all have their own focus areas with respect to logging, it should be possible to provide a solution that fulfils all or most requirements. Such a solution must be open and extendable, for example, by using a plug-in framework such as in Pitch Recorder. An initial demonstrator based on the Pitch Recorder plug-in framework has been described in this paper and has shown that a new data stream such as SIMPLE/Link 16 can be added relatively easily to the Pitch Recorder.

One important conclusion is the need to record several types of data in parallel to fully capture the exercise in particular in LVC and training applications. This may include both standardized data streams, like HLA, DIS and voice as well as proprietary data.

Future work on data logging and playback, in particular work related to debrief, should not only consider the requirements listed in paper, but also look at the work of the SISO Distributed Debrief Control Protocol (DDCP) Study Group [15]. The aim of the DDCP Study Group is to evaluate industry and government interest in developing a distributed debrief control protocol standard. Some of the requirements in this paper are related to this work.

REFERENCES

[1] A Henninger, et. al. "Live Virtual Constructive Architecture Roadmap (LVCAR) Final Report", US DoD, September 2008.

[2] IEEE: "IEEE 1516-2010, High Level Architecture (HLA)", www.ieee.org, August 2010.

[3] SISO, "Real-time Platform Reference Federation Object Model 2.0 ", SISO-STD-001 SISO, draft 17.

[4] IEEE: "IEEE 1278, Distributed Interactive Simulation (DIS)", www.ieee.org.

[5] Link 16 is defined as one of the digital services of the JTIDS / MIDS in NATO's Standardization Agreement STANAG 5516. MIL-STD-6016 is the related United States Department of Defense Link 16 MIL-STD.

[6] "TENA - The Test and Training Enabling Architecture, Architecture Reference Document", https://www.tena-sda.org/public_docmanager/userdocuments/TENA ARCHITECTURE REFERENCE/TENA Architecture Reference Document_2002.pdf.

[7] Metropolis, N. and Ulam, S. "The Monte Carlo Method." J. Amer. Stat. Assoc. 44, 335-341, 1949.

[8] 08E-SIW-003: Application of HLA in the Optimization of Rail Transport. Euro SIW 2008. T.W. van den Berg et al.

[9] 09S-SIW-008: Execution Management Solutions for Geographically Distributed Simulations. Spring SIW 2009. T.W. van den Berg et al.

[10] 11E-SIW-010: Generic Reconstruction and Analysis for simulations or live exercises. Euro SIW 2011. R. Witberg et al.

[11] Pitch Recorder web page, http://www.pitch.se/products/recorder

[12] Viking 11, http://www.forsvarsmakten.se/en/About-the-Armed-Forces/Exercises/Completed-exercises-and-events/VIKING-11/

[13] Standard Interface for Multiple Platform Link Evaluation (SIMPLE). STANAG 5602 (Edition 2). http://nsa.nato.int.

[14] Joint Project Optic Windmill, http://www.globalsecurity.org/military/ops/optic-windmill.htm.

[15] SISO Distributed Debrief Control Protocol(DDCP) Study Group, http://www.sisostds.org/Standards Activities/StudyGroups DDCPSGDistributedDebrief ControlProtocol.aspx.

This article is reprinted from the Fall 2011 Simulation Interoperability Standards Organization (SISO) Simulation Interoperability Workshop (SIW), paper 11F-SIW-055.

REBUILDING THE NAVSEA EARLY STAGE SHIP DESIGN ENVIRONMENT

Ben Kassel
Naval Surface Warfare Center, Carderock Division

Seth Cooper
Naval Sea Systems Command

Adrian Mackenna
Naval Surface Warfare Center, Carderock Division

ABSTRACT

It has been well over a decade since the last time the Naval Sea Systems Command (NAVSEA) performed a full blown preliminary and contract ship design. During that time period there have been many advances in the underlying technology used by design tools, and there have also been changes to the design process as well. As a result, NAVSEA has the responsibility to evaluate tools and processes in order to develop the next generation early stage ship design environment so that we do not continue to design tomorrow's ships with yesterday's tools. This paper discusses the role product model technology, high performance computing, and early stage design tools can play in the development of future naval vessels. The subject of design tools will be explored from the perspective of how they improve the early stage ship design process as well as their role in gaining insights and supporting oversight during the detailed design and construction phases of a ship's lifecycle.

KEYWORDS

- **CAD** - Computer-Aided Design
- **CREATE** – Computational Research and Engineering Acquisition Tools and Environments
- **HPC** – High Performance Computers
- **ONR** – Office of Naval Research
- **FEM** – Finite Element Model
- **FEA** – Finite Element Analysis
- **IGES** – Initial Graphics Exchange Specification
- **NPDI** – Navy Product Data Initiative
- **STEP** – Standard for the Exchange of Product data (ISO 10303)
- **LEAPS** – Leading Edge Architecture for Prototyping Systems
- **ASSET** – Advanced Ship and Submarine Evaluation Tool
- **COTS** – Commercial Off-the-Shelf Software

INTRODUCTION

Change is a permanent fixture within the US Naval Shipbuilding industry, to the acquisition process, and within the NAVSEA enterprise. It has been several years since an early stage design has been led and completed by the government (Keane et al. 2009). Major changes have occurred in both the sophistication of software products available to the marine industry as well as the available computing power. Open architectures and the availability of standards for the definition of product model data has the potential to improve the early stage design process. Of course, many issues arise when establishing a design site, but this paper only examines issues of product model technology, software, and early stage design tools. But one thing is for sure, the early stage knowledge embedded within the NAVSEA enterprise is retiring. The humans that managed, performed, and supported early stage ship design are all but gone. If the next generation of early stage ship designers are not deliberately trained, mentored, and given the tools they need to design 21st century ships within the next few years, there is a distinct possibility there will be none of the current generation left to pass on the trade.

A BRIEF HISTORY

Up through the 1990's design sites supporting the early stage design of surface ships and submarines were commonplace within NAVSEA (Ayers et al. 1998). These design sites could be found within NAVSEA office spaces, contractors' facilities, and at the Naval Ship R&D Center in Carderock, MD. They were staffed by a mixture of NAVSEA and Warfare Center employees and resources obtained from local Naval Architecture firms. Depending on the acquisition strategy, some of the design teams would include the shipyards that may be bidding on the detailed design and construction. During these bygone days, NAVSEA was deeply involved in the use and customization of commercial CAD systems, the continuous evaluation of commercial

Naval Architecture tools, and where necessary, the development of specific design tools. NAVSEA, as well as any major enterprise involved in the design of their products struggled with the balance between the use of commercially available software and the in-house development of software. This problem was compounded by the relatively small size of the marine sector coupled with trying to define the Navy's core competencies in software development. During the acquisition reform of the mid-nineties, many of the responsibilities traditionally assigned to the NAVSEA engineering directorate were transferred to the industrial sector. One critical result was less Navy engineering and more Navy engineering oversight. The old adage goes "you forget what you hear, you remember what you see, and you know what you do." Because NAVSEA is not doing ship design, it is missing an opportunity to pass on corporate engineering knowledge to the next generation of ship designers, ship design managers, and design integration managers. The time is ripe for NAVSEA to rebuild its early stage design competency. With improvements in information technology, we are afforded an opportunity to integrate cutting edge information technologies with established analysis tools and the knowledge of an aging workforce.

THE CASE FOR TOOL DEVELOPMENT

Ships are large and complex products and have a long development cycle. It is widely recognized throughout the engineering world that decisions made during the conceptual design phase have the largest impacts on cost, performance, and schedule. Many of the critical requirements levied on a warship require complex analysis to verify that they are met (such as hull fatigue life, vulnerability/shock performance, signatures, and topside sensing/communication performance). These complex analyses require a high level of design definition, which is typically not available until the detailed design and construction phase. In the current design paradigm, analysis results that verify if a ship design meets its requirements come after their opportunity to influence the design. Because of the limited amount of tool integration, and a manual ship design definition process, the Navy enterprise usually driven to select one design alternative early in the design process. Much of the rest of the design effort is spent detailing and reworking this single alternative to meet the requirements and cost goals.

The vision for Navy design tools is to move to a automated high-end toolset that integrates many information dense design definition tools with high fidelity physics-based analysis tools. This toolset will be able explore many ship design alternatives to populate a feasible design space. This design space will be used to perform real time cost-benefit trades on ship requirements during the requirements definition process. A system such as this could be used to explore the design space to ensure that the correct design is selected before signing a contract to build a ship.

Direction on this was given to NAVSEA in February of 2008 in a memo from Admiral Sullivan, who was then COMNAVSEA, which outline types of tools and tool developments needed. The memo stated that,

> Accomplishing these ambitious goals will be a challenge, but is essential for crafting affordable, executable ship programs in an increasingly complex national security environment. Previous Navy design tool investment has resulted in the Advanced Ship and Submarine Evaluation Tool (ASSET) for total ship synthesis, and the Leading Edge Architecture for Prototyping Systems (LEAPS) for integrating a wide range of analysis tools in a common data environment. Future tool development should build upon these foundations, adding capability to meet the goals outlined in this memorandum. [Ser 05D/047, 4 Feb 2008]

ASSET is software that has been built and maintained by the Navy at NSWC Carderock for over 25 years, and it is currently the principal tool used in earliest stages of ship design. ASSET is unique in that it combines ship design disciplines into one synthesized whole-ship model that represents a balanced design. A major issue is that ASSET does not produce the level of design definition required for many of the higher-level analyses required in the later stages of ship design. When a design progresses beyond concept design, where a more detailed analysis is needed, the design integration provided between disciplines by ASSET is lost. Existing analysis tools typically require their own custom format of input data. Up to 90% of the time spent on these analyses is spent preparing the input, which often means manually recreating design data that was already created in another tool. This data recreation accounts for most of the time, cost, and error associated with analysis.

The effort to solve this time-delay and configuration control issue between high-end tools is LEAPS. LEAPS is also

developed and maintained by the Navy at NSWC Carderock, and has been a 15-year effort. At its base LEAPS is a digital representation of the ship designed to be expansible to include all information necessary to perform any relevant analysis and store the results of those analyses for use by other analyses. It is the hub, while detailed discipline-specific tools represent the spokes in a ship design cycle. Careful though and planning is required to bring a LEAPS based design and analysis for a new ship to fruition. What design and analysis activities are performed at each phase of the design should be planned carefully to ensure that the information is created before it is required. The way that design tools interact is directly related to the way we design ships, and the way people think about design. The process that forms the foundation of ASSET reflects the roles and responsibilities of NAVSEA at the time ASSET was created. Efforts are currently underway to map the entire ship design process so that gaps in the ship design toolset can be identified. This map will also allow NAVSEA to engineer and streamline the ship design process. This effort is tied to the NAVSEA Tools Roadmap development and semi-annual workshops sponsored by NAVSEA, ONR, and the CREATE program, but this is a subject for another paper.

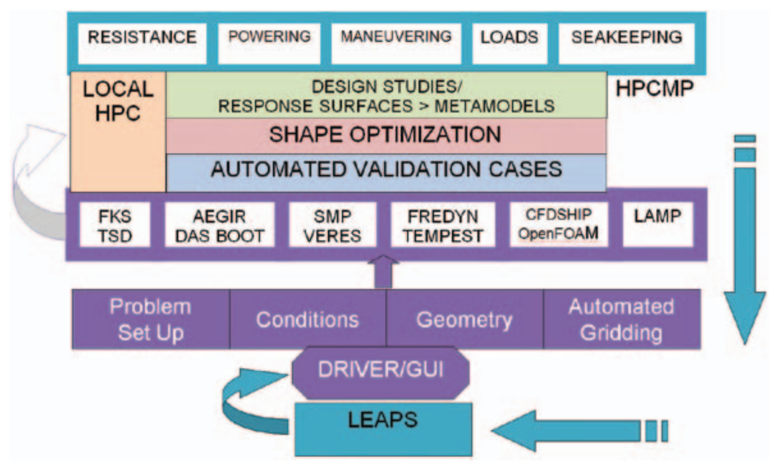

Figure 1 The Integrated Hullform Design Environment ties together several hydrodynamics tools

CREATE is a DoD program that is focused on tackling many of these challenges. The program is run out of the DoD High Performance Computers Modernization Office (HPCMO). CREATE-SHIPS (a portion of the overall CREATE program) is budgeted to spend several million dollars from HPCMO over the next few years, and focuses on leveraging the modern increases in computational power to develop the high-end toolset and enable this process of rapidly designing and analyzing large number if ship designs. CREATE-SHIPS is a partnership between NAVSEA, ONR, HPCMO, and PEO SHIPS. Another positive step at NAVSEA was the establishment of the Technical Warrant Holder position for Ship Design Tools in October of 2008 as a step towards raising the importance of ship design tools in the overall design and certification process. In December 2009, this position was moved to the NAVSEA 05 Chief Technology Office as the Tools Program Manager, further elevating the importance of tool development to the NAVEA enterprise.

In addition to the issue of configuration between disciplines, many aspects of ship design do not have sufficient tools and models in existence, and increasingly rely on engineering judgment and large factors of safety. For instance, we are often looking at new and innovative ways to estimate a ship's manning requirements or costs at an early stage. But developing and improving the individual high-end tools themselves is not as simple as implementing a theory into a computer algorithm. Tools need to be verified and validated; problems must be easy to set up and run; geometry and mesh generation must be easy and quick; tools must be built to run effectively and efficiently on highly complex massively parallel computers; and, results must be timely. Many of the tools we use are highly specialized, and not used beyond the narrow realm of Naval ship design. The results of these complex analyses must be visualized and packaged in a way that they are easy to understand by both the design engineers and program managers such that they can be the basis of a smart, timely decision making process.

A CREATE effort of note is the development of the Integrated Hydrodynamics Design Environment (IHDE). For ship hydrodynamics a large set of specialized commercial, government, and even university built research tools and models is used for all aspects of ship hydrodynamics such as resistance, seakeeping, stability, and fluid-structure interactions. Most of these tools are highly specialized and only experts can run them. The IHDE, now in its second year of development, seeks to provide a unified easy-to-use system that gives a ship designer the ability to interface more directly with these tools. It also has the ability to create

input files from ship design data available in the LEAPS representation of the ship automatically, and to store the results of the analysis back into LEAPS.

Another software development worthy of discussion here is Intelligent Ship Arrangements (ISA), which is a tool in its infant stages developed as a research project at the University of Michigan and not yet transitioned to Navy use. As mentioned earlier, many cases an analysis cannot be performed due to a lack of the design definition needed. A major hurdle is that ship arrangements—the way that compartments and machinery are laid out relative to one another—is an intensive manual process and often considered more art than science because of the unlimited number of viable solutions. This tool looks at the arrangements within a ship's hull as an industrial engineering problem, a hybrid of efficiently packing a box and laying out circuitry on a microchip, and arranges the ship according to constraints and rules set by the users ahead of time. When used in a systematic and stochastic way, and when integrated using LEAPS, having this type of design information early in the design process can feed into analyses such as manning, vulnerability, producibility, and a number of other "ilities" in time to influence major ship design decisions.

Ultimately, our goal is to shrink the time required to generate a sufficient amount of information to make informed design decisions early in the ship design process before the requirements are set and cost of the ship is locked in. By considering an integrated computational ship model as a "virtual prototype," several design iterations are possible in a far shorter amount of time than a single design-build-test cycle of a traditional prototype.

One commercial example illustrates what we can now achieve with this paradigm. In the early 1990s, Goodyear Tire faced intense international competition. Its rivals had more engineering design resources, testing capacity, and lower production costs—Goodyear was rapidly falling behind. To respond and develop a competitive advantage, it replaced the traditional engineering process (design, build, test, and repeat) that had served it well for more than 100 years with physics-based computational engineering tools to design, mesh, and analyze new products. Engineers built and tested just the final, optimized designs, thereby reducing Goodyear's time to market from three years to less than a year. The company started producing several new designs a year instead of one or two every few years. Goodyear is now the largest US tire manufacturer and is competitive in the world market. Whirlpool, Proctor and Gamble, Boeing, Ping Golf, and Pratt and Whitney, to name a few, have also adopted this new paradigm with similar success.

In addition, Systems Engineering tools and methodologies such as Set-Based Design along with techniques for Design of Experiments and Multi-Disciplinary Optimization can help integrate seemingly disparate types of analysis. Stochastic analysis, now available to us through automation and high-speed computing, will not only allow us to better capture uncertainty into the design process, but it allows several single aspects of a ship design to be explored comprehensively on their own before comparing them to ensure convergence and feasibility of the ship design as a whole. In addition to linking ship structures, hydrodynamics, and susceptibility models for instance, the front end can link to force models and the back end can link with cost and affordability models to provide a full picture to decision makers so that timely decisions can be made with confidence.

AN INTRODUCTION TO EARLY STAGE SHIP DESIGN

Decisions made early on in the ship design process have large impacts on ship functionality that isn't quantified until the design is mature. Often these impacts are only vaguely understood at the outset of the design cycle, and by the time that the impacts are fully understood it is too late to make significant changes. An example of this could be the vulnerability of the ship. In order to asses a ship's vulnerability, a detailed layout of compartments and distributed systems is needed. However, early on in the ship design, when sizing decisions are made, detailed layouts are not available. A ship designer has little more than rules-of-thumb on which to base these crucial decisions. With High Performance Computing (HPC) as an enabler, the vision is to explore all downstream implications of decisions made during the initial concept development and apply that knowledge as early on in the design process as possible. In the vulnerability example used above, for instance, an automated tool (such as ISA mentioned earlier) could rapidly produce a full range of feasible ship arrangements from a basic shell of a ship. Then a vulnerability assessment could be performed on each of these many design variations and the resultant range of achievable levels of vulnerability can be fed back to the designer—with all of the high-speed computation happening behind the scenes. Thus, the designer is instantly aware of the vulnerability implications of the sizing and arrangement of the ship.

DESIGN SPIRAL VERSUS SET-BASED DESIGN

Naval Ship Design involves complex interactions between many disciplines, and reconciling the needs of one system against others becomes a delicate balancing act. The convergence of various discipline-specific ship models into a single coherent design is a process that NAVSEA has termed "Ship Synthesis," and is currently chiefly performed using the Navy's in-house tool ASSET. ASSET is made up of discipline specific modules (i.e., hull geometry, gross arrangement, hull structural design, resistance and propulsion, power plant sizing, weight estimation, and area/volume sufficiency analysis). ASSET performs synthesis between these modules using a design spiral approach. This means that disciplines are analyzed one at a time before moving to the next one, and multiple iterations are performed through the spiral process in order to converge into a single solution. Each loop is a serial process that must be done in order, and control of each design variable must be carefully managed. The modules in ASSET are highly coupled so that the dynamic process of synthesis is stable and converges on a solution.

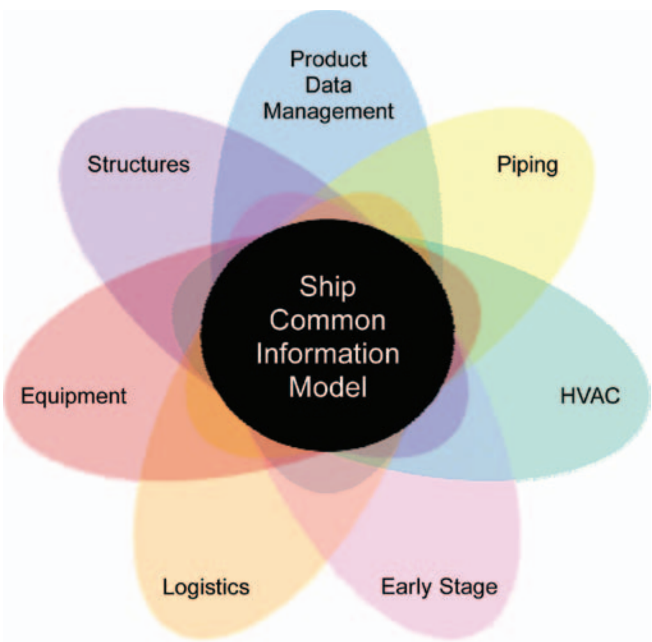

Figure 4 - Ship Common Information Model

In a Set-Based Design approach, which has been identified as a preferred approach for the development of future U.S. Naval design efforts, discipline-specific designs are done in parallel across a broad design space. This process is designed to improve the flexibility of the design by delaying key decisions until the design space is fully understood, but the parallel nature of the approach also makes it an ideal fit for HPC application. Currently, set-based design, as practiced in the Navy's Ship-to-Shore Connector program, relies on engineering interaction and judgment for creating the set information from each discipline and integrating the results from multiple disciplines in order to find a set of possible solutions. But techniques known as Multi-Disciplinary Optimization (MDO) offer the infrastructure for integrating the set based design theory into Navy ship design tools in a mathematically rigorous and automated manner. Applications of MDO techniques to ship synthesis are ready to be tested and implemented and are moving forward. Due to the highly coupled, multivariate nature of the ship design problem, MDO will be challenging, but holds great promise as an integrating agent.

THE NAVY BUSINESS MODEL

It is important at this point in the discussion to recognize the business environment in which the Navy designs ships. Although the details of ship procurement seem to change weekly, the basic initial design process remains generally the same. In the initial phase a large design space of many options is explored to a low level of fidelity, and with each successive phase of the design, guidance is given to the designer by a decision maker and the fidelity of the design is increased along with a decrease in the range of options available. This process is depicted graphically in Figure 2. As the design space starts to become defined, higher order models can be substituted into the Ship Synthesis process. HPC tools, such as the ones being developed under the CREATE Hydro and Shock projects, can become appropriate higher order models when the design space becomes sufficiently defined, and as these tools become faster and more accessible, they can be used in earlier phases of the design.

HPC tools for other disciplines, beyond just hydrodynamics and shock, need development as well, and an analysis of which disciplines hold promising theories that are applicable to solution by HPC (i.e., large amounts of numerical calculations) should be done, and investments should be made in those areas. An example of this is the Intelligent Ship Arrangement technique under development at the University of Michigan, which was mentioned earlier in the paper.

ASSET EVOLUTION

The vision for future ASSET is to expand its ship definition and analysis capability. Ship definition capability will be greatly enhanced through the use of a 3D NURBS based geometry definition/manipulation, arrangements, and component placement capability. This capabiltiy will come through the conversion of the ASSET data model into a LEAPS data model. This will allow the rapid turnaround between this design tool and higher fidelity physics models that require complicated mesh-able geometry and detailed ship design information. Arrangement details such as topside design and distributed system routing/architecture will make it possible to assess topside effectiveness and vulnerability during concept design. Pre-defined components will be stored in a LEAPS database and used in the ASSET model. ASSET will be run in a batch mode to create hundreds or thousands of feasible design variants that will be analyzed to determine their effectiveness.

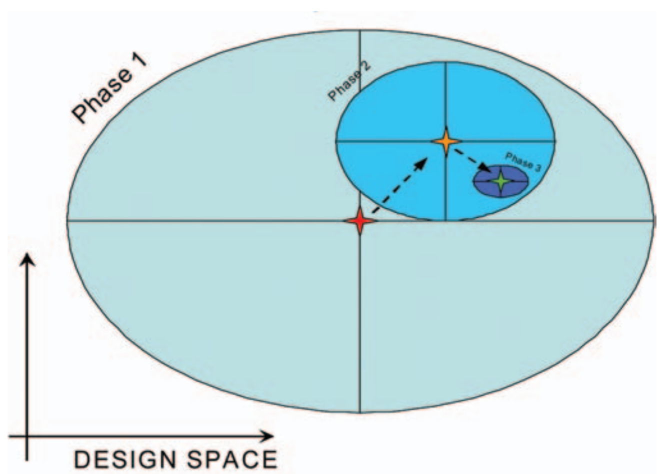

Figure 2 - As the fidelity of the design increases, the design space under consideration gets smaller.

USING BEHAVIOR MODELS

In order to use higher order models within a Ship Synthesis process and get results in a timely manner, the results of many runs of the higher order models must be abstracted into a "Behavior Model." These are sometimes also referred to as "Surrogate Models," or "Response Surfaces." The idea behind a Behavior Model is that from many discrete points in a design space, a continuous function can be closely fit, and that function can be queried instantaneously rather than re-running the computationally intensive higher order model. In this way, the Ship Synthesis process using full physics models can be done in real time. The design space can be explored or an optimization performed in a reasonable timeframe by a single user. Done in this manner, several higher order models in several disciplines can be run in a highly parallel way over a broad range of design space prior to the Ship Synthesis process. Figure 2 illustrates the process of defining a feasible design space in an initial phase of the design. Later an aspect of the design is explored in more detail using HPC tools, and a Behavior Model is developed from the results. This Behavior model can then be incorporated back into a Ship Synthesis in the next design phase. A similar approach to this was taken during the ONR HSSL effort, where seakeeping and resistance Behavior Models were developed based on parametric hullform changes, and these Behavior Models were used as part of Ship Synthesis. Using HPC, this effort was able to build these Behavior Models in two to three days rather than the 5,000 plus hours of computing time that it would have taken otherwise.

Several mathematical models exist for developing Behavior Models, including polynomial splines, neural networks, and Kriging, and some are more appropriate than others for certain applications. These methods need to be characterized and developed into software that works within the Navy's ship design infrastructure and can be used in a parallel computing environment.

LEADING EDGE ARCHITECTURE FOR PROTOTYPING FOR SYSTEMS

LEAPS is a Navy developed environment for storing information about ship designs. It functions as a database that is capable of storing multiple ship concepts, including detailed geometry information, numerical design information, system attribution, and behavior objects. A further discussion of LEAPS as a product model is discussed later in this paper.

Currently, CREATE Ships has funded the University of Michigan to update the database structure of LEAPS to enable queries to be made in parallel, enabling the use of LEAPS in a highly parallel computing job such as running many designs through multiple scenarios in a seakeeping analysis.

The DOD CREATE program, along with NAVSEA, is also currently developing the capability within LEAPS

to parametrically distort a parent hullform so that a large design space can be created and run through HPC analysis tools such as those accessible from the Integrated Hydrodynamics Design Environment. This hullform manipulation toolkit is also an enabler for doing set-based design of the hullform in parallel with other aspects of the ship.

PLANNED TOOL IMPROVEMENTS FOR THE CONCEPT DESIGN PHASE

In the concept design phase, the ship design organization should explore large sets of potential design alternatives using design space exploration and visualization methods. To use this method, an automated toolset is needed that can rapidly populate a design space with performance, cost, and risk data. There are several areas of improvement that have been identified to fill in a complete toolset for concept design. As we continue to identify and prioritize these, actions should be taken to improve those areas.

During this phase the level of detail may be relatively low, but the design is extremely dynamic. The process is dominated by synthesis tools and low level analysis. The emphasis during this phase is to identify solutions that are feasible. The majority of the design effort is performed using ASSET and a host of analysis tools that will quickly at a low level of detail identify windows of feasibility considering many variables including: cost, weight, arrangeable space, powering, and many others (Doerry 2009). The plan is to improve ASSET, build/integrate additional design and analysis tools, and provide a tighter integration with LEAPS (NAVSEA 05D 2008).

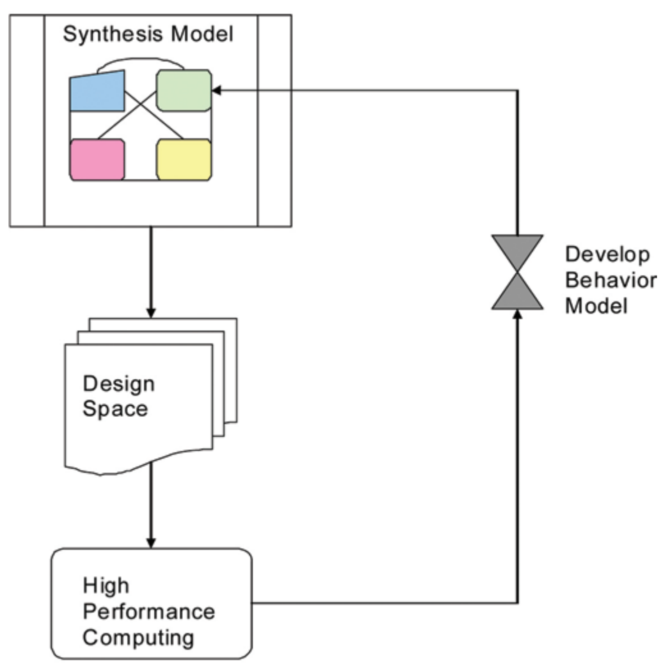

Figure 3. Higher order models used in successive design phases in the Ship Synthesis process can be Behavior Models created from HPC results.

There are several products that are planned. These include Force Architecture Assessment and Operational Effectiveness Analysis. Plans are to integrate all of the design information into the LEAPS schema.

PRELIMINARY/CONTRACT DESIGN

In the preliminary design phase, ship designers will continue to explore sets of potential design alternatives, but now to a higher level of fidelity and a less broad range. Design integration of a set based preliminary design will be challenging. Design space visualization is required to understand the whole ship impact of design decisions. As mentioned above, ASSET will be further integrated with LEAPS, so that higher fidelity preliminary design information can replace the lower fidelity concept design information initially generated by the program. The program will also allow multiple users to work in parallel on different parts of the ship to speed the design definition process. In this phase of the design, the synthesis process will be much more focused on individual aspects of the design, which will be worked in great detail, whereas larger scale changes will be less common.

The Graphical User Interface (GUI) for ASSET will need to be much improved. ASSET of the future will feature a GUI that guides the user through the design process. The new GUI will allow the user to have point-and-click subdivision definition and interactive arrangement capability that will allow the user to see and manipulate the design in three dimensions. This GUI will allow the user to place machinery components, place topside equipment, and define distributed system runs using a three dimensional representation of the ship. The ability to place topside equipment on the three dimensional product model of the ship will allow ASSET to perform topside design. ASSET will also be able to interface with physics based topside analysis tools using a LEAPS interface. A topside design utility will feature a complete library of existing topside sensors, where pertinent design information has been pre-populated. The user will be able to define

the necessary distributed system runs that will allow the ASSET model to be used to populate the data necessary for vulnerability analysis. This will enable vulnerability analysis earlier in the design process. The ASSET GUI will allow the user to do general arrangements of the ship design, enabling the functional allocation of space to be made in the ship design process. This capability will allow the user to consider modularity during the design, and quantify how the placement of modules affects the general arrangement. Automated internal and topside arrangement capability is currently under development in the Intelligent Ship Arrangement tool.

A mission system component catalog will be developed. This data will be captured in a LEAPS database of group 400 (sensors and communications equipment) and group 700 (weapons) systems that are found in existing ships, or being considered for future ships. The pedigree of the information will be stored in the LEAPS database along with each component to indicate if the system attributes are "as built" or were captured at some "non-final" stage. This database will be accessible and used as a primary payload database for ASSET. The component catalog will contain the component attributes necessary to determine the ship impact and perform the necessary analysis. The information will contain weight, area/volume, power, cooling, component specific location restrictions, and the specifics required for analysis (such as electromagnetic emissions). This data will be mined from certified data sources, and will, therefore, become the authoritative data set that can be referenced and used for future design studies. A process will be put into place to guide a user through the proper population of a mission system into ASSET. A mission system configuration utility will be added to ASSET to guide the user through a selection and placement process.

Functional Arrangement

During the preliminary design a greater emphasis is placed on optimizing the general arrangement by considering the location of equipment, outfitting, and routing lanes. The first iteration of the preliminary design LEAPS model is a product of the ASSET synthesis developed during the concept phase. The current process is heavily dependent upon commercial CAD tools where the geometry manipulation capabilities of the CAD system can be used to detail the arrangement. This process is heavily dependent on the existence of a reliable and efficient data exchange capability. It is envisioned that a limited portion of the arrangement function will be performed in an automated manner using the Intelligent Ship Arrangements (ISA) tool.

This evolving functional arrangement continues to be closely coupled to the hullform, many times requiring analyses typically associated with the concept phase. During this phase the level of detail is increasing, and more importantly the ship design is maturing. This enables more types of analysis to be performed in order to validate that the design meets the requirements. These types of analysis include; stability, vulnerability, survivability, shock, and a more in-depth evaluation of structural strength and fatigue performance, and hydrodynamic performance.

COTS VS. ORGANIC

This question has been haunting NAVSEA forever. During the mid 1980's it became especially contentious in an era nostalgically referred to as "the CAD wars." One faction was adamant that the only way NAVSEA could obtain design tools, including drafting tools was to have them developed in-house. The other faction was equally as adamant that the CAD industry could provide all tools necessary to support early stage ship design. We have since learned that a combination of COTS and organic design tools are necessary; although, we are not always properly performing this trade-off. We have also learned that most of the time COTS tools require some amount of customization to be useful for Navy design applications. The reality is that even the organic tools require a formal set of processes to ensure that they are used correctly. If a COTS package has the capability required, can reasonably be integrated into the design process, and proves to be the most cost effective solution, it should be used. NAVSEA has limited resources to develop and maintain in-house software, and it should be used to develop core Navy capabilities that commercial industry has no incentive to develop on its own.

INTEGRATING DESIGN TOOLS

The Navy's plan is to implement LEAPS as the method for integrating the design information, and in many cases, interacting design tools. There are two options to using LEAPS as the design tools integrator. Option one is to modify design tools to directly use the LEAPS repository as their native database format. Option two is to create a translator that will extract that information required for performing the

analysis and upon completion of the analysis will write the results back to the LEAPS repository. The first option is best for new tool development, while the second option may be more palatable for existing tools.

LEAPS AS A SOFTWARE ENVIRONMENT

LEAPS is a very powerful software environment that includes a CAD and math engine, and several useful toolkits. Both the LEAPS software and all of the supporting documentation is now approved for unlimited distribution, and is available to software developers for free if they want to use and even distribute it with their software. In return, the Navy hopes to have available many more software tools with the innate ability to extract and save ship data to LEAPS files. This year, we will be making LEAPS even more accessible by creating a web-based community of developers, where questions and examples can be exchanged, and LEAPS software and documentation downloaded. In addition, we continue to expand the user community of LEAPS through navy sponsored software development, where use of LEAPS is made contractual.

PROVIDING INFORMATION TO "THE OUTSIDE"

Digital product models have the ability to provide much more information about a ship than paper drawings. Information on design intent, engineering analyses, and inter-relation of systems can all be presented together, but as a technology the Navy is still learning to use it effectively. The issue, as we have learned, is that digital data formats come and go, and the lifetime of a CAD system is often shorter than the lifetime of a class of ship. Whereas, storing paper drawings amounts to a fairly trivial task, storing digital data has issues with computer systems, operating systems, and software systems that are all constantly changing and evolving. LEAPS provides the opportunity for the Navy to be stewards of their own digital data. Rather than relying on the constantly changing tide of the commercial sector, by defining and maintaining the LEAPS standard, the Navy can ensure that its own digital data stands the test of time. The optimal solution is a neutral file format that is not only product model agnostic, but transcends all phases of the ships lifecycle. The reality is that a compelling case can be made for archiving the native data along with one or more neutral representations. The key lies in having a thorough understanding of the context in which each format has the advantage and the pedigree of the information while maintaining strict configuration control. (Rakow, et. al. 2009) Surely maintaining LEAPS as a standard for ship design data must be recognized as a Navy core capability.

One of the major technical hurdles will be the method for providing this information to prospective bidders. From the perspective of the Navy, the easiest path would be to provide access to the LEAPS repository on a Navy controlled Integrated Product Data Environment. Another option worthy of exploration is to provide that information in a standards based neutral format. At this time the leading candidate for a neutral approach makes use of the Standard for the Exchange of Product Model Data (STEP). Although this may be the major milestone that would require digital data to be provided by the Navy, it is not the only time. The exchange of digital data is something that will be performed in a near continuous fashion to support collaboration during a lifecycle phase.

OBTAINING INFORMATION FROM "THE OUTSIDE"

Data exchange is required in both directions, especially to support a collaborative effort. During the Preliminary and Contract Design phases, there is a high probability that information will be required which has been developed outside of the NAVSEA design tools environment. Not only may it be useful to obtain data that may have been developed and created by shipbuilders, using their production oriented tools, but from a myriad of other sources as well. These data sources may include equipment suppliers, weapons system integrators, and as we migrate to an "open architecture," the pool of qualified suppliers will expand significantly. In recognition of this environment, NAVSEA and the commercial shipbuilders through the National Shipbuilding Research Program are working to identify the minimum set of information that needed to define a ship and ships systems. This Ship Common Information Model (NSRP 2008) is a multidisciplinary view of product model data and transcends life cycle phases as shown in figure 4. It is envisioned that this view will be developed in collaboration by NAVSEA 04, NAVSEA 05, the shipbuilders, and suppliers of design tools and Product Data Management tools. The owner of a specific piece of the Ship Common Information Model may have their own requirements but the content will be balanced since the stakeholders span the entire ship lifecycle.

The data obtained from the outside concentrates on the as-designed arrangement. Typically, NAVSEA would like to obtain the geometry and associated non graphical data in suitable detail and format to enable independent

analysis to validate that the design meets the requirements. This means, in addition to the as-designed arrangement, NAVSEA will look to the shipbuilder to provide design data such as the (a) molded forms suitable for defining a general arrangement, (b) scantling level of detail of structure to support structural (and other types of) analysis, (c) functional distributed systems model (i.e., path, components, and connections), (d) compartmentation, including accesses, opening, and tightness, and (e) some fundamental equipment properties (i.e., weights, centers, electrical loads). The availability of this data is a key element in enabling NAVSEA technical warrant holders and engineers to operate within the NAVSEA 05 tools environment, in accordance with VADM Sullivan's vision as stated in the 2008 memo (ref: Ser 05D/047 dtd 4 Feb 2008). It will also provide accurate data of value to the NAVSEA 04 community as they prepare to provide support for ships after they are delivered to the fleet.

DETAILED DESIGN

As a ship design progresses, the responsibility of "detailed design" is handed off to industry, and the types of models used for the design, primarily CAD and manufacturing models, are fundamentally different than the physics-based ship performance models used within the Navy. At the time the detailed design contract is awarded, the physics–based analysis to ascertain whether the ship meets its requirements should be complete, and the ship design configuration should be fixed. A crucial challenge will be the ability to translate the shipyard's detailed design data back into a digital format appropriate for meshing and analyzing the performance of the designs. The lifetime of a ship class from the time the lead ship is conceived to the time the final ship is retired far exceeds the lifetime of any commercial ship design or CAD tool, and yet computer-based analysis is needed throughout the ships' life for refits, upgrades, or damage incidents. It is crucial as well that the Navy become stewards of the digital ship design data for their assets. These are two more reasons why the Navy must continue to not only build and maintain the LEAPS system, but enforce its use.

The Navy does not do detailed design of ships, but during detailed design the Navy has a continued responsibility to be a smart customer by a continuous process of accepting data for review and performance analysis. As the design matures, NAVSEA does not need manufacturing data, but does need geometry structure, arrangements, and parts catalog data for systems and payloads. The component catalog data that is captured in LEAPS for new class specific systems will then be available for future preliminary designs, and can be easily accessed to assess commonality of future designs with those of the past.

THIS IS ONLY THE BEGINNING

This paper discussed the emerging tools, modeling, and product data integration environment being developed to support early stage naval ship design. It is true that Naval ship design was performed well before any of the advanced computational capabilities we seek today were available, but with NAVSEA at less than a quarter of the size it was in the eighties, the rising cost of ships, and the increasing complexity of technology, we cannot afford to not have the most powerful tools available. Unfortunately, this is balanced by the current budget for tool development also standing at about a quarter of what it was in the eighties (not adjusted for inflation). So the challenge grows tougher as we continue to develop tomorrow's ships with yesterday's tools.

REFERENCES

[1] Sullivan, P., "Ship Design and Analysis Tool Goals," NAVSEA Memorandum Ser 05D/047, February 2008.

[2] Keane, R., J. McIntire, H. Fireman, and D. Maher, "The LPD17 Ship Design: Leading a Sea Change Toward Collaborative Product Development", Naval Engineers Journal, Vol. 121, No 2, pp15-61, 2009.

[3] Benthall, L., T. Briggs, B. Downie, B. Gischner, B. Kassel, and R. Wood, "STEP for Shipbuilding: A Solution for Product Model Data Exchange", SNAME Journal of Ship Production, Vol. 19, No 1, pp44-52, February 2003.

[4] Kassel, B., T. Briggs, "An Alternate Approach to the Exchange of Ship Product Model Data", SNAME Journal of Ship Production, Vol. 24, No 2, pp92-98, May 2008.

[5] Ayers, R., P. Callahan, B. Kassel, "Application of a General Purpose CAD System in the DDG-51 Design Process," Naval Engineers Journal, Vol. 100, No 3, pp265-274, May 1988.

[6] Kassel, B., "The Use of CAD in the Development of an Engine Room Arrangement Model," Application of Information Technologies to the Maritime Industries, MAREXPO Consortium, pp217-232, June 1999.

[7] Rakow, P., Kassel, B., Mays, J. "Long Term Data Retention of Weapon System Information Project Status report," NSWCCD 2230, November 2009.

[8] NSRP ASE, NSRP Navy Product Data Initiative Integrated Product Data Environment (IPDE) Specification", June, 30 2008.

[9] Doerry, N. CAPT, USN, Steding M., "Ship Design Manager Manual," Naval Sea Systems Command, June 1, 2009.

[10] NAVSEA 05D, "Future Concepts and Surface Ship Design Design Tools Roadmap," Naval Sea Systems Command, 2008.

[11] Miller, Loren K., "Competitive Advantage in New Products through Simulation-Based Engineering," DataMetric Innovations, 2009.

This article is reprinted from the American Society of Naval Engineers (ASNE) Day 2010.

ABOUT THE M&S JOURNAL

THE M&S JOURNAL is a theme-based, quarterly publication of articles that highlight M&S technology, applications, prototype processes or products, points of view, or emerging philosophies. The M&S Journal is a valuable resource for the M&S Community: across DoD, other government agencies, international partner organizations, industry, and academia. All submitted technical papers for the M&S Journal undergo rigorous peer review following an initial screening for conformance to basic requirements. Publishing in the M&S Journal affords authors both an online and print forum for their M&S technical papers, gaining recognition and publicity throughout the M&S Community. Authors also receive extended visibility for their ideas through free online access to their article on the Modeling and Simulation Coordination Office (M&SCO) website. http://www.msco.mil

SUBSCRIPTIONS: If you would like to subscribe to the M&S Journal, please send an email to: journal-subscribe@lists.dod-msiac.org.

SUBMISSION GUIDELINES

SUBMISSIONS must be previously cleared material for open distribution, and must include a reprint permission statement.

They may be entirely new work, or previously published papers that would benefit from a wider exposure and would provide valuable resources for M&S users.

MANUSCRIPTS should be between 3000 and 9000 words and should be submitted in Microsoft Word format.

The M&S Journal Editorial Staff reserves the right to modify a paper for the purpose of typographical or grammatical corrections.

The author will be notified whether the submission has met the basic requirements for the M&S Journal, and will be notified again when the final acceptance/rejection decision has been made.

The M&S Journal does not accept papers that are structured as commercial advertising, or as promotions of products or services.

If you would like to submit a technical paper for consideration to the Journal, please e-mail it together with contact information to:

MSIACHelpDesk@DoD-MSIAC.org

SPRING EDITION 2012
VOLUME 7 ISSUE 1

THE M&S JOURNAL

A PUBLICATION OF M&SCO
THE DOD MODELING & SIMULATION COORDINATION OFFICE